



**A DISCRETE-EVENT SIMULATION MODEL FOR EVALUATING AIR FORCE  
REUSABLE MILITARY LAUNCH VEHICLE PRELAUNCH OPERATIONS**

**THESIS**

**Adam T. Stiegelmeier, Captain, USAF**

AFIT/GLM/ENS/06-16

**DEPARTMENT OF THE AIR FORCE  
AIR UNIVERSITY**

**AIR FORCE INSTITUTE OF TECHNOLOGY**

---

---

**Wright-Patterson Air Force Base, Ohio**

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the U.S. Government.

AFIT/GLM/ENS/06-16

A DISCRETE-EVENT SIMULATION MODEL FOR EVALUATING AIR FORCE  
REUSABLE MILITARY LAUNCH VEHICLE PRELAUNCH OPERATIONS

THESIS

Presented to the Faculty

Department of Operational Sciences

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the  
Degree of Master of Science in Logistics Management

Adam T. Stiegelmeier, B. A.

Captain, USAF

March 2006

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

A DISCRETE-EVENT SIMULATION MODEL FOR EVALUATING AIR FORCE  
REUSABLE MILITARY LAUNCH VEHICLE PRELAUNCH OPERATIONS

Adam T. Stiegelmeier, B. A.  
Captain, USAF

Approved:

//signed//  
Alan W. Johnson, Ph.D. (Chair)

\_\_\_\_\_  
date

//signed//  
Sharif H. Melouk, Ph.D., (Member)

\_\_\_\_\_  
date

## **Abstract**

As the control and exploitation of space becomes more important to the United States military, a responsive spacelift capability will become essential. Responsive spacelift could be defined as the ability to launch a vehicle within hours or days from the time a launch order is given, instead of the weeks or months it takes currently. As the Air Force contemplates moving toward a reusable military launch vehicle (RMLV) capability, it faces key design and ground processing decisions that will affect the vehicle regeneration timeline. This thesis develops a computer simulation model that mimics RMLV prelaunch operations—those activities that take place during vehicle integration and launch pad operations. This simulation model can help the Air Force make RMLV acquisition decisions by analyzing how different RMLV designs and ground processing scenarios will affect RMLV regeneration time.

The model was developed by comparing and contrasting existing launch vehicle processing flows to create the RMLV prelaunch operations model. To foster confidence in model credibility, the model was analyzed and validated by a panel of launch vehicle experts. Model verification was accomplished via an Assertion Checking method that compared model developer intent to actual model operation. The model was used to conduct three experiments that analyzed how different ground processing scenarios affected RMLV regeneration time.

## **Acknowledgments**

I would like to thank my faculty advisor, Dr. Alan Johnson, and reader, Dr. Sharif Melouk for their guidance and help throughout this research process. Additionally, I would like to thank my sponsor, Bruce Thieman, from the Air Force Research Laboratory's (AFRL) Operationally Responsive Space Office, for providing TDY funds and access to information. Alexis Larson and Tom Jacobs, both from the AFRL Air Vehicles directorate, also provided key contacts and information. Special thanks also goes out to Robert Johnson (Kennedy Space Center) and Brendan Rooney (Aeronautical Systems Center) for all the helpful information and advice. I also appreciate the efforts of each member who participated in the model review Delphi process.

My family deserves special recognition. They have earned this degree right along with me! To my wife—thank-you for your patience, understanding, and support. To my children—thank-you for the joy you brought me when the research got tough. I love all of you very much.

Finally, and most importantly, I want to thank my Lord, Jesus Christ, for His great love and grace. He is the one who ultimately allows me to accomplish anything, and I am thankful for His help and guidance throughout this research.

Adam T. Stiegelmeier

## Table of Contents

	Page
Abstract.....	iv
Acknowledgments.....	v
List of Figures.....	viii
List of Tables.....	ix
I. Introduction.....	1
Background.....	1
Problem.....	3
Research Objective.....	5
Summary and Preview.....	6
II. Literature Review.....	7
Introduction.....	7
Air Force and NASA Future Spacelift Objectives.....	7
Significance of RMLV Prelaunch Operations Research.....	9
Length of Shuttle Prelaunch Operations.....	10
Length of Expendable Launch Vehicle Prelaunch Operations.....	12
RMLV Alert Capability and Prelaunch Operations.....	22
Previous Launch Vehicle Simulation Research.....	23
Prelaunch Operations Comparisons.....	26
Shuttle Comparison.....	27
Aircraft Comparison.....	33
ELV Comparison.....	36
ICBM Comparison.....	37
Summary.....	42
III. Methodology.....	43
Introduction.....	43
Applicability of Discrete Event Simulation to the Research Problem.....	43
Model Development.....	45
The 12-Step Modeling Process.....	46
Step 1: Problem Formulation.....	46
Step 2: Setting of Objectives and Overall Project Plan.....	47
Step 3: Model Conceptualization.....	47
Step 4: Data Collection.....	47
Step 5: Model Translation.....	49

	Page
Step 6: Verification.....	50
Step 7: Validation.....	51
Step 8: Experimental Design.....	54
Step 9: Production Runs and Analysis.....	55
Step 10: More Runs?.....	56
Step 11: Documentation and Reporting.....	56
Step 12: Implementation.....	56
Summary.....	57
IV. Results and Analysis.....	58
Introduction.....	58
Model Description.....	58
Model Verification Results.....	72
Experimental Design Results.....	78
Experiment 1: Effect of Preintegration Decision.....	80
Experiment 2: Effect of Integration Location Decision.....	81
Experiment 3: Effect of Integration Orientation Decision.....	83
Summary.....	85
V. Conclusions and Recommendations.....	86
Introduction.....	86
Research Summary.....	86
Research Conclusions.....	87
Research Limitations.....	91
Suggestions for Future Research.....	93
Appendix A. Delphi Study Documentation.....	96
Appendix B. Graphical User Interface Forms.....	123
Appendix C. Prelaunch Operations Activity Duration Distributions.....	131
Bibliography.....	132
Vita.....	136



## List of Figures

	Page
Figure 1: Time Spent on Shuttle Vehicle Integration in Vehicle Assembly Building .....	11
Figure 2: Time Spent on Shuttle Launch Pad Operations .....	11
Figure 3: Atlas V .....	14
Figure 4: Delta IV .....	15
Figure 5: Delta IV Medium Launch Vehicle CCAFS Projected Processing Timeline ....	16
Figure 6: Atlas V Processing Activities Prior to Launch Day .....	17
Figure 7: Atlas V Launch Day Activities .....	17
Figure 8: Zenit 3SL.....	20
Figure 9: Zenit 3SL atop the Floating Launch Platform.....	21
Figure 10: Shuttle Prelaunch Operations .....	28
Figure 11: Orbiter being mated to the rest of the space shuttle vehicle .....	29
Figure 12: Space shuttle vehicle being transported to the launch pad.....	29
Figure 13: Payload installation operations at the launch pad .....	30
Figure 14: Notional RMLV Design .....	31
Figure 15: Notional RMLV Design .....	32
Figure 16: Minuteman III .....	37
Figure 17: Transporter Erector .....	38
Figure 18: Minuteman III Reentry System Install .....	39
Figure 19: Technicians Working from Elevator Workcage .....	41
Figure 20: The 12 Steps in a Simulation Study .....	46
Figure 21: Graphical user interface user form example .....	59
Figure 22: Module Key.....	62
Figure 23: Preintegration .....	63
Figure 24: Vehicle integration preliminary decisions .....	64
Figure 25: On-pad integration operations.....	65
Figure 26: Off-pad integration operations .....	66
Figure 27: Initial launch pad operations for vehicle integrated off pad.....	67
Figure 28: Launch pad operations .....	68
Figure 29: Propellant loading operations.....	69

## List of Tables

	Page
Table 1: HLV OS Operational Requirements.....	9
Table 2: RMLV design and processing decisions .....	73
Table 3: Entity path for first verification test .....	74
Table 4: Record module expected versus actual values .....	75
Table 5: Selected reliability acceptance plan.....	77
Table 6: Entity paths compared for the preintegration experiment .....	80
Table 7: Preintegration experiment output statistics .....	81
Table 8: Entity paths compared for the integration location experiment .....	82
Table 9: Integration location experiment output statistics.....	83
Table 10: Entity paths compared for the integration orientation experiment.....	84
Table 11: Integration orientation experiment output statistics .....	85

# A DISCRETE-EVENT SIMULATION MODEL FOR EVALUATING AIR FORCE REUSABLE MILITARY LAUNCH VEHICLE PRELAUNCH OPERATIONS

## I. Introduction

### Background

The control and exploitation of space is becoming increasingly important for the United States military. The Department of Defense relies heavily on capabilities provided by space assets, so much so that a disruption of those capabilities would have grave implications. Many of the technologies utilized during Operation Iraqi Freedom (OIF), such as global positioning system guided munitions and satellite surveillance, depend upon the military's uninhibited use of space. So far the military's use of space has not been significantly challenged, but this could change in the future. One high-ranking Russian military official advocated the development of Russian antisatellite weapons in a post-OIF assessment of U. S. military capability (Brown, 2004b:1). It is likely that U. S. space assets will become increasingly vulnerable as other nations improve their space access capabilities.

To maintain U. S. superiority in space, the Air Force needs *responsive* spacelift capability that will enable quick access to space for national defense purposes. Responsive spacelift could be defined as the ability to launch a vehicle within hours or days from the time a launch order is given, instead of the weeks or months it takes currently (Brown, 2004b:2). The Air Force is using the term Operationally Responsive Space to describe the responsive spacelift concept. Replacing or repairing damaged

satellites are key capabilities enabled by Operationally Responsive Space. Other possible Operationally Responsive Space applications include facilitation of future space missions such as space control, missile defense, and force application (Brown, 2004b:5).

The Air Force currently utilizes a family of expendable launch vehicles (ELV) to meet its spacelift needs. Unfortunately, this method of space access is far from responsive. It commonly takes weeks or even months to prepare an ELV for launch. Furthermore, each launch is extremely expensive since all vehicle hardware is essentially “thrown away” after each use. For example, it is estimated that each launch of the Titan IVB ELV cost between \$350 and \$450 million (Isakowitz et al, 2004:496). Launch costs for other ELVs vary, but many are in the hundreds of millions of dollars. In an effort to reduce launch costs, the Air Force is pursuing the development of reusable military launch vehicles (RMLV). It is hoped that a fleet of RMLVs will significantly reduce launch costs because each vehicle would be refurbished after its mission and used again. The Air Force is currently in the research and development phase of obtaining a vehicle with a reusable first stage and expendable second stage. This particular vehicle concept has been termed the Hybrid Launch Vehicle (HLV S&A SOO:1). The Hybrid Launch Vehicle recurring flight cost goal is to reduce launch costs to 1/6 of the current ELV launch costs (HLV S&A SOO:4).

The only operational orbital reusable launch vehicle (RLV)<sup>1</sup> in the world is the space shuttle. While a significant technological achievement in its own right, the shuttle cannot be categorized as responsive spacelift. On average, it takes 126 calendar days to

---

<sup>1</sup> The acronym RLV is used to refer to non-military reusable launch vehicles, like the space shuttle. RMLV is used to refer to reusable launch vehicles used for military purposes.

regenerate the shuttle (McClesky, 2005:29). Regeneration includes all inspection, maintenance, and servicing activities undertaken between vehicle landing and the next vehicle launch. The Air Force wants the capability to regenerate the Hybrid Launch Vehicle in 24 hours or less (HLV S&A SOO:4).

## **Problem**

It is obvious that the Air Force must design an RMLV that facilitates regeneration times that are much shorter than the regeneration times experienced by the space shuttle. To put it another way, the Air Force's RMLV fleet must experience much higher availability levels than that of the space shuttle fleet. Availability is defined as "the probability that a component or system is performing its required function at a given point in time when used under stated operating conditions" (Ebeling, 2005:6).

Mathematically it can be represented by

$$Availability = \frac{uptime}{uptime + downtime} \quad (1)$$

where uptime is the amount of time spent in an operable state and downtime is the amount of time spent in an inoperable state. Availability can be increased in several ways. First, increases in vehicle reliability will increase availability. Reliability is defined as "the probability of non-failure over time" (Ebeling, 2005:5). Reducing the probability of component failure will decrease the amount of time spent on reactive maintenance, or fixing components that break. It can also reduce the amount of time spent on proactive maintenance, which includes inspecting or replacing components to keep them from failing. Increases in maintainability will also increase availability.

Maintainability is defined as “the probability of repair in a given time” (Ebeling, 2005:6).

Maintainability improvements include easier access to components and increased fault isolation capability; they generally help maintenance personnel return the vehicle to serviceable status more quickly (Ebeling, 2005:223).

One aspect of RMLV ground operations closely related to maintainability is vehicle handling and servicing. Just as increases in maintainability speed up vehicle recovery, increases in vehicle handling and servicing efficiency can decrease the amount of time it takes to prepare a vehicle for launch. For RMLVs, handling and servicing include vehicle transport, stage mating and payload integration, and servicing of fuel and other fluids and gasses. These actions are often loosely spoken of as prelaunch operations, because they follow other vehicle preparation steps and occur immediately prior to vehicle launch. Brown divides these actions into two categories: call-up time and launch operations. He defines call-up time as “the time required to prepare for launch, starting with the vehicles in standby mode in a hangar and ending with the payload integrated into the vehicle and ready for launch operations on the launch pad.” He defines launch operations as “all activities on the launch pad beginning with propellant and pressurant loading and ending with the engine start command” (Brown, 2004a:11). To avoid confusion and wordiness, this thesis will use the term “prelaunch operations” to describe the actions of payload integration, stage mating, vehicle transport, and vehicle servicing. Prelaunch operations start when vehicle assembly (stage mating and payload integration) begins and end when the vehicle’s engines ignite. The assumption is that at the start of prelaunch operations, all major repairs and inspections are complete and that apart from the prelaunch steps, the vehicle is operable. It is also assumed that the

payload and all stages are completely processed and ready for immediate assembly. The prelaunch processing sequence is of key importance because it has the potential to add a significant amount of time to RMLV ground operations. For the Air Force to reach its RMLV turn-around time goal, it must place special emphasis on utilizing efficient prelaunch operations methods.

### **Research Objective**

The purpose of this research is to aid the Air Force in its search for efficient prelaunch operations by creating a discrete-event simulation model of a generic RMLV prelaunch process. Such a model can help decision makers evaluate tradeoffs between vehicle design alternatives and prelaunch operations efficiency. It can also give insight into the prelaunch steps that add the most time to the prelaunch process. To guide the research effort, the following research question is proposed:

How can the Air Force develop a discrete-event simulation model of RMLV prelaunch operations that will aid decision makers in evaluating RMLV design alternatives?

The research question is divided into the following investigative questions:

1. What generic functions, or sequence of actions, describe RMLV prelaunch operations?
2. How do these RMLV prelaunch operation functions compare to shuttle, aircraft, ELV, and Intercontinental Ballistic Missile (ICBM) prelaunch operation functions?

3. What are the RMLV design drivers that will influence RMLV prelaunch operations, and how will these drivers affect the number, type, and duration of RMLV prelaunch operations activities?
4. How can these RMLV design drivers and prelaunch operations activities be incorporated into a discrete-event simulation model that captures a baseline RMLV prelaunch operations sequence?
5. What RMLV regeneration timeline insights can be gained from running the model using notional but plausible inputs?

## **Summary and Preview**

This chapter provided a general overview of the need for responsive spacelift and described the challenges to achieving responsive spacelift. A definition of prelaunch operations was presented. The purpose for this research was discussed, along with the overall research question and investigative questions. The following chapters will address the answers to these questions. Chapter II will provide an overview of general prelaunch operations, emphasize the importance of efficient prelaunch operations, and discuss previous launch vehicle simulation models. Chapter III will describe the methodology used in this research. Chapter IV will include a description of the model developed for this thesis along with model verification and experimental design results. Chapter V will offer research conclusions and future research ideas.



## **II. Literature Review**

### **Introduction**

This chapter will provide a more detailed background of the research problem by explaining key terms and concepts, justifying why the problem is significant, and discussing the research that has been accomplished to date. The first section will cover the future spacelift objectives for both NASA and the Air Force. This will be followed by a general discussion of prelaunch operations for current space launch systems to give the reader a sense of the importance and extent of the problem. The next section will include an overview of the existing research and simulation models pertaining to RLV ground operations. The chapter will conclude with a section that will compare notional RMLV prelaunch operations to similar shuttle, aircraft, ELV, and ICBM activities.

### **Air Force and NASA Future Spacelift Objectives**

To begin, it will be helpful to understand the current context of both Air Force and NASA spacelift objectives. In January of 2004, President Bush released his new vision for the nation's space exploration program. The vision outlines the nation's space exploration goals for the next several decades and mandates a retirement of the space shuttle by the year 2010. The space shuttle will be replaced with a new space vehicle that will be used for manned lunar missions and eventually manned missions to Mars. To this end, NASA is focusing its resources on development of the Crew Exploration Vehicle ("President Bush Announces," 2004). Crew Exploration Vehicle plans call for manned missions beginning no later than 2011 (NASA's Exploration Systems Architecture Study,

2005: 13). The Crew Exploration Vehicle will likely be launched via a shuttle Solid Rocket Booster for the first stage and a yet-to-be-designed liquid fueled second stage (NASA's Exploration Systems Architecture Study, 2005:42, 43).

The Air Force's spacelift needs differ from NASA's spacelift needs. NASA is the "civil" space agency for the U. S. and often flies for exploratory or research purposes according to a fixed launch schedule. In contrast, the Air Force is a military organization requiring a spacelift capability that can quickly respond to war-time contingencies. The Air Force is considering several uses for its future space launch capability, including satellite replacement; intelligence, surveillance, and reconnaissance; and rapid global strike (Wall, 2002:39). As a step towards developing a responsive spacelift capability, Air Force Space Command initiated an Analysis of Alternatives study in March of 2003. The purpose of the Analysis of Alternatives study was to analyze different approaches to achieving Operationally Responsive Space and to recommend a specific acquisition strategy for design and fielding of an Air Force responsive spacelift capability (Dornheim, 2003:70). The Analysis of Alternatives study recommended a design composed of both reusable and expendable elements. This design has been termed the Hybrid Launch Vehicle. The Hybrid Launch Vehicle will be made up of a reusable first stage and one or two expendable upper stages. The Analysis of Alternatives study chose the Hybrid Launch Vehicle concept because it lent itself to quick turn-around times and was generally less expensive than fully reusable or fully expendable options. The Hybrid Launch Vehicle is currently in the concept development and demonstration planning stage. A contract will be awarded at the end of 2007 for a Hybrid Launch Vehicle subscale demonstrator that will undergo ground and flight tests by 2011. The actual

operational vehicle, or Hybrid Launch Vehicle Operational Spacelift (HLV OS), is scheduled to begin operations by 2018. The Air Force will require the HLV OS to be highly responsive, with a goal of launching a pre-integrated payload with a 24 to 48 hour notice (Dornheim, 2005:33). Other HLV OS requirements and goals are given in Table 1.

**Table 1: HLV OS Operational Requirements (HLV S&A SOO: 3)**

<b>Operational Parameter</b>	<b>Threshold</b>	<b>Objective</b>
<b>First Stage Turn -Around Time</b>	<b>48 hours</b>	<b>24 hours</b>
<b>HLV OS Recurring Flight Cost</b>	<b>1/3 current EELV -M launch costs</b>	<b>1/6 current EELV -M launch costs</b>
<b>HLV OS Initial Production Size</b>	<b>6 Operational First Stages</b>	<b>6 Operational First Stages</b>
<b>First Stage Return to Base (RTB) – Nominal Mission</b>	<b>Required</b>	<b>Required</b>
<b>First Stage RTB – Intact Abort</b>	<b>50%*</b>	<b>90%*</b>
<b>Blue Suit Operators</b>	<b>Blue Suit &amp; Contractor</b>	<b>Blue Suit</b>
<b>HLV OS Upper Stages Production Costs</b>	<b>\$10M per unit</b>	<b>\$5M per unit</b>
<b>Use of Foreign Designed Critical Components</b>	<b>Domestic Production Required</b>	<b>No Foreign Designed Components</b>

### **Significance of RMLV Prelaunch Operations Research**

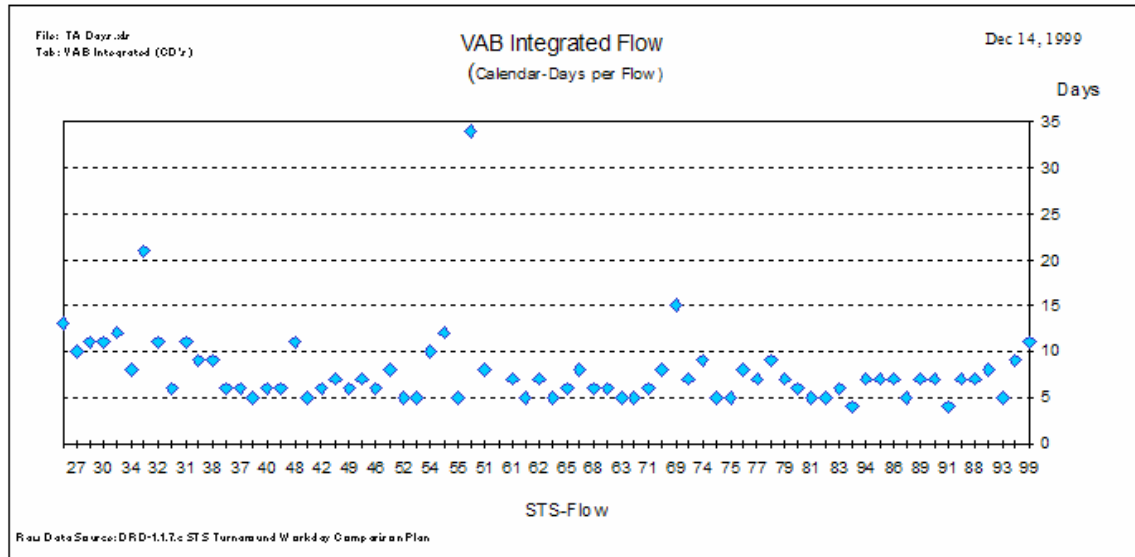
There may be a tendency to discount the importance of research that seeks to develop timely prelaunch operations for RMLVs. It is true that most of the time spent on vehicle regeneration will likely be spent on the maintenance functions that precede prelaunch operations. Because of this, improving the timeliness of these maintenance functions may have the most benefit for improving regeneration time overall. However, this does not negate the importance of timely prelaunch operations. As demonstrated in Figures 1 and 2, drastic improvements in the timeliness of the core maintenance functions preceding prelaunch operations with no or little improvement in the timeliness of

prelaunch operations over existing launch systems will still leave the Air Force short of its RMLV regeneration goals. Furthermore, as the space community continues to emphasize “aircraft-like” RMLV operations, RMLV prelaunch operations become especially pertinent since such operations will by necessity differ greatly from similar aircraft operations. The following sections attempt to illustrate the importance of developing timely RMLV prelaunch operations by discussing the unresponsive nature of prelaunch operations for existing systems and by showing how RMLV alert operations will be especially dependent upon a rapid RMLV prelaunch sequence.

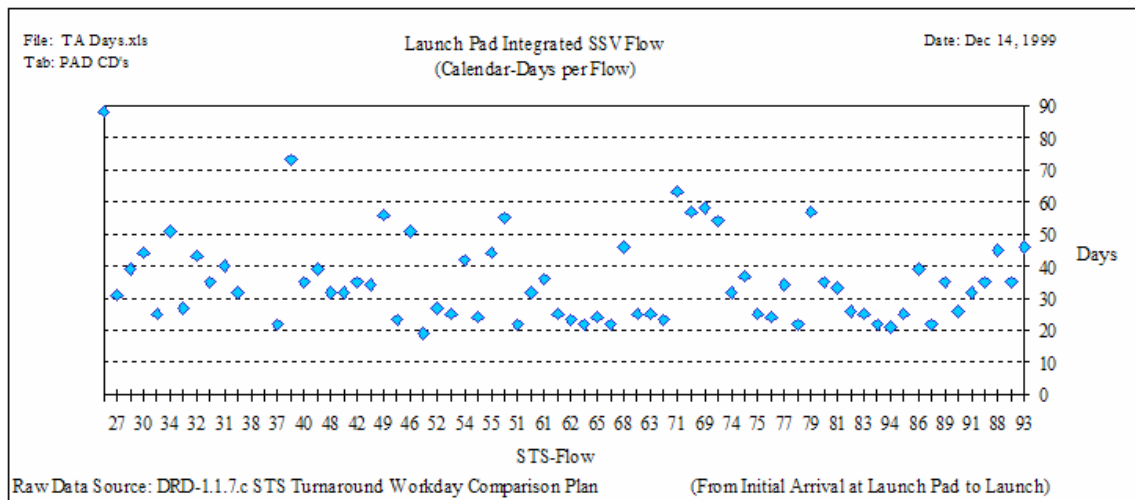
### ***Length of Shuttle Prelaunch Operations***

The shuttle experience demonstrates how much time prelaunch operations can add to total regeneration time. The shuttle, which is the only operational orbital RLV, takes on average approximately 126 days to regenerate (McClesky, 2005:29). But how much of this total time is taken up by prelaunch operations? Before answering this question, it will be helpful to describe the actions that make up shuttle prelaunch operations. A more detailed description of shuttle prelaunch operations is given in the “Prelaunch Operations Comparisons” section below, and for now, a brief explanation will suffice. In the introduction, prelaunch operations were defined generically as all actions accomplished from vehicle integration to engine ignition. For the shuttle, this would include Vehicle Assembly Building operations, transport to the launch pad, and launch pad operations. The Vehicle Assembly Building is where the major vehicle components are joined together on top of the Mobile Launch Pad. After Vehicle Assembly Building processing is complete, the shuttle is transported via crawler transporter to the launch pad, where final checks, crew ingress, and propellant servicing take place (Cates, 2003:89-115).

Historical data on shuttle Vehicle Assembly Building time and launch pad time is displayed graphically in Figures 1 and 2.



**Figure 1: Time Spent on Shuttle Vehicle Integration in Vehicle Assembly Building (TA Days)**



**Figure 2: Time Spent on Shuttle Launch Pad Operations (TA Days)**

On average, shuttle integration in the Vehicle Assembly Building takes approximately seven calendar days (Cates, 2003:112,113).<sup>2</sup> Transport to the launch pad takes approximately one day (Cates, 2003:117). The average shuttle launch pad processing time is approximately 35 calendar days (Pad SSV TA Days). Added together, these operations account for approximately 43 days of total shuttle regeneration time. Or to put it another way, approximately one-third (43 days out of 126 days) of shuttle regeneration time is taken up by prelaunch operations activities. This data clearly demonstrates the importance of timely prelaunch operations. For the Air Force to reach its RMLV turn-around time goals, it must obtain a vehicle that will allow for much quicker assembly, transport, and launch pad operations.

### ***Length of Expendable Launch Vehicle Prelaunch Operations***

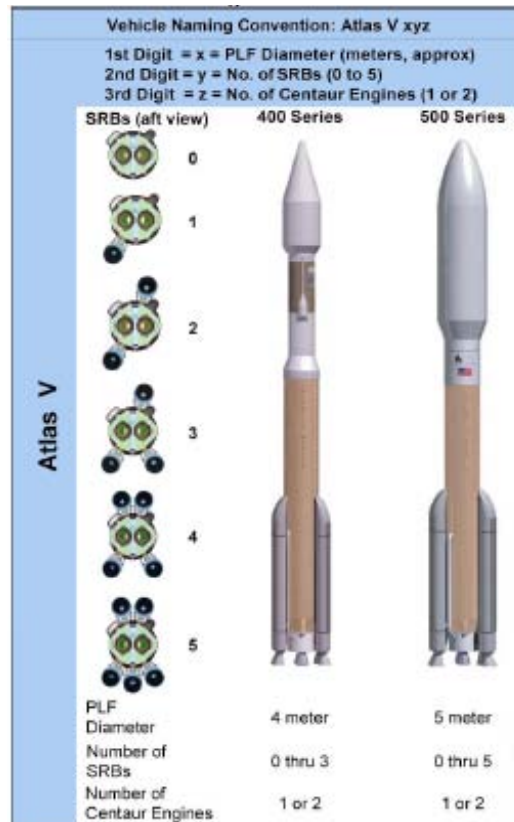
Ever since the Space Shuttle Challenger accident, the military has relied exclusively on ELVs to place payloads into orbit (Greaves, 1997:9). A wide range of different ELVs are in operation today, but the most applicable group to our study is the Evolved Expendable Launch Vehicle family. The Evolved Expendable Launch Vehicle family is composed of both Delta IV and Atlas V launch vehicles. An outgrowth of older ELV programs, the Evolved Expendable Launch Vehicle represents advances in technology and ground operations over its predecessors and is thus the most pertinent, up-to-date example to consider for this study.

The exorbitant costs associated with ELV launches motivated the effort to field a more cost-effective and reliable launch capability in the mid-1990's. The Evolved

---

<sup>2</sup> These seven calendar days do not include solid rocket booster stacking operations and external tank-to-solid rocket booster connections. Solid rocket booster stacking and external tank-to-solid rocket booster connections occur prior to the mating of the orbiter to the rest of the space shuttle vehicle and, if included, would increase total integration time.

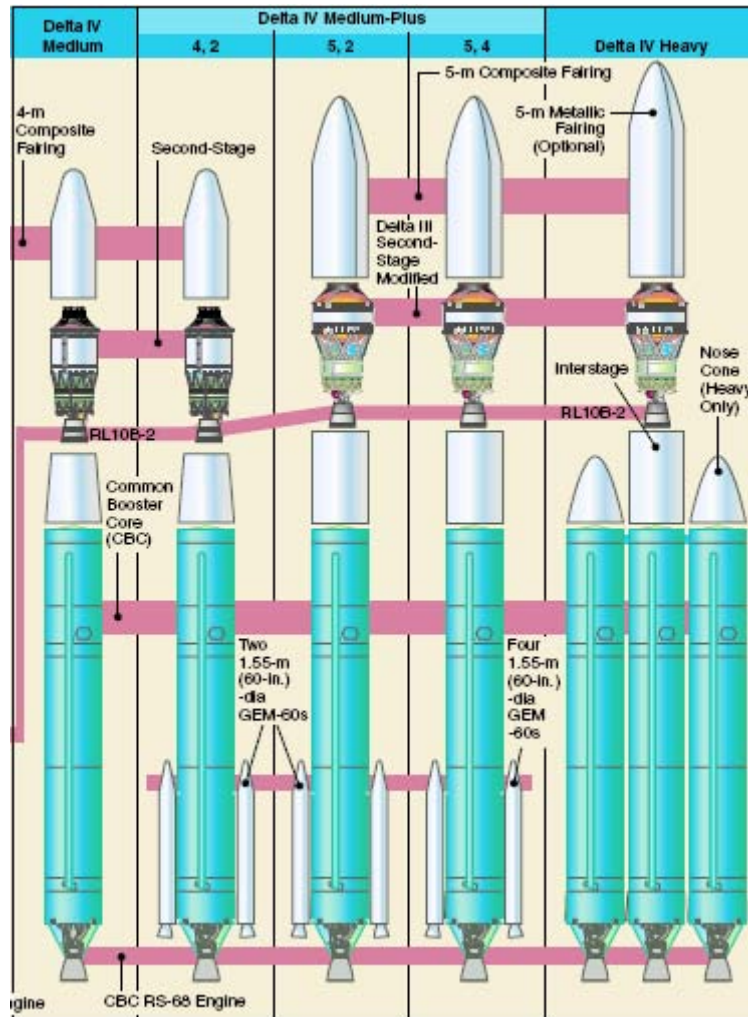
Expendable Launch Vehicle was the result of this effort. The Evolved Expendable Launch Vehicle is truly “evolved” in the sense that both Evolved Expendable Launch Vehicle variants, Atlas V and Delta IV, are based upon older, proven ELV technology. The Atlas V is a Lockheed Martin product and is an outgrowth of the Atlas II and Atlas III. The Delta IV is produced by Boeing and is based upon the Delta II and Delta III (Isakowitz et al, 1999:128). The Evolved Expendable Launch Vehicle program was initiated with several key objectives. First, the Evolved Expendable Launch Vehicle program was expected to preserve the capability already achieved by existing launch systems to safely and accurately place satellites in orbit (Greaves, 1997:31). Second, the Evolved Expendable Launch Vehicle program was designed to reduce overall recurring launch costs by 25 to 50 percent over existing launch systems (GAO, 2004:1). Finally, the Evolved Expendable Launch Vehicle program was expected demonstrate “operability improvements” over older launch systems (Evolved Expendable Launch Vehicle, 2005:par. 1). The Atlas V and Delta IV launch vehicles are shown in Figures 3 and 4, respectively. While some have questioned whether or not the Evolved Expendable Launch Vehicle program will be able to fully demonstrate its expected cost savings (GAO, 2004:7-9), the Evolved Expendable Launch Vehicle has successfully incorporated ground operations changes that allow for more timely prelaunch operations. For instance, both the Atlas V and Delta IV are designed to minimize the amount of time the vehicle spends on the launch pad. Delta IV is integrated and tested horizontally in a horizontal integration facility. Once all systems are verified for launch, the vehicle is transported horizontally to the launch pad where it is erected and then integrated with its payload (Delta IV Launch Vehicle). This is an improvement over Delta II, its



**Figure 3: Atlas V (Atlas Launch System Mission Planner’s Guide, 2004:1-4)**

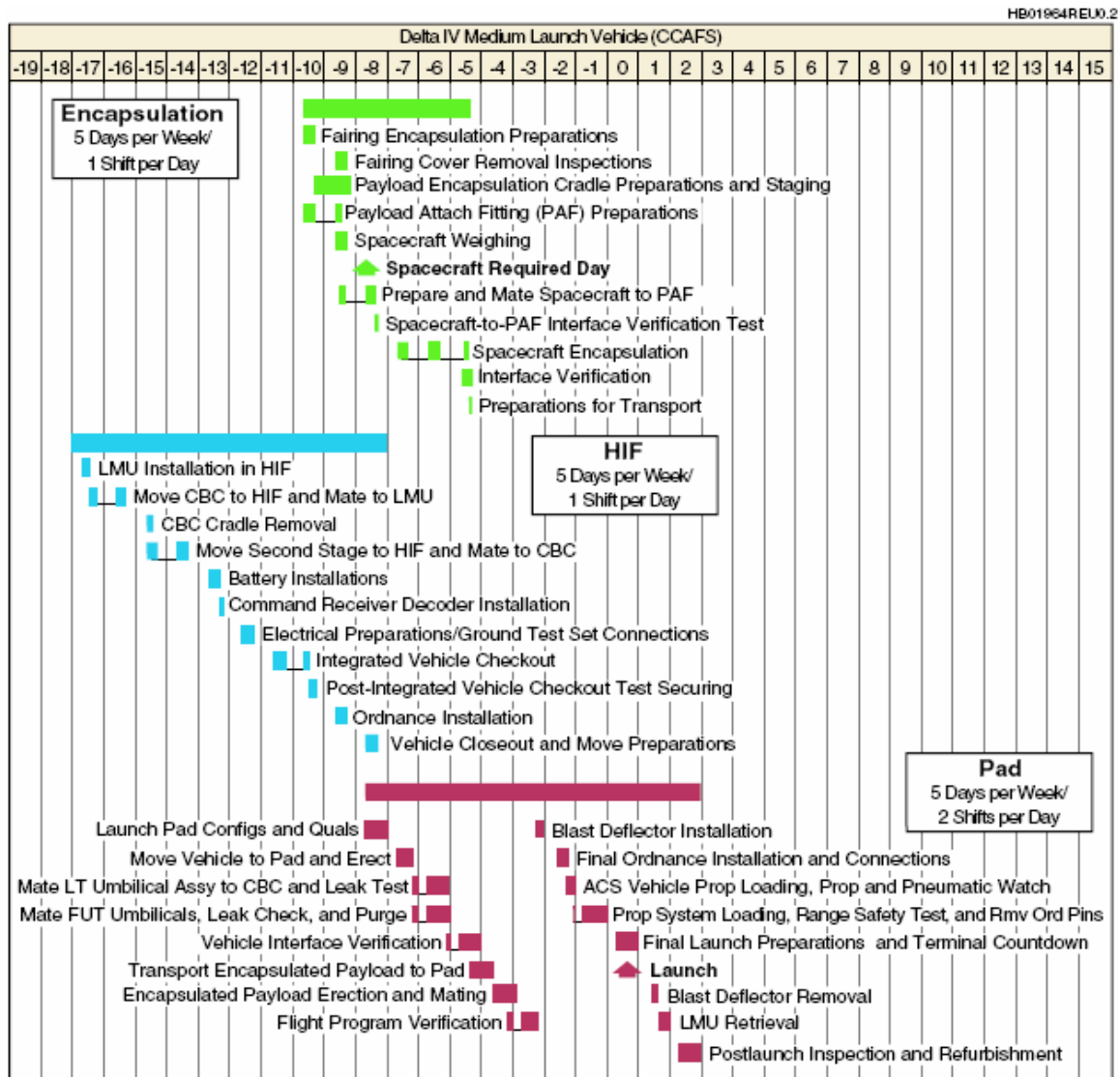
predecessor, which undergoes all integration and testing while on the launch pad, significantly increasing the duration of launch pad operations (Isakowitz et al, 2004:139). Likewise, at the Cape Canaveral Air Force Station launch site, Atlas V undergoes all stage and payload integration activities off the pad in a vertical integration facility and is moved to the launch pad eight and a half hours before the scheduled launch (Centore, 2005:15). Atlas V also significantly speeded its task documentation process by implementing an electronic documentation tool that replaces traditional paper documentation procedures. Many launch processing tasks for Atlas V, such as propellant loading, are accomplished automatically via a computer monitored process. Taking out the “human element” in this way streamlines tasks and eradicates processing time





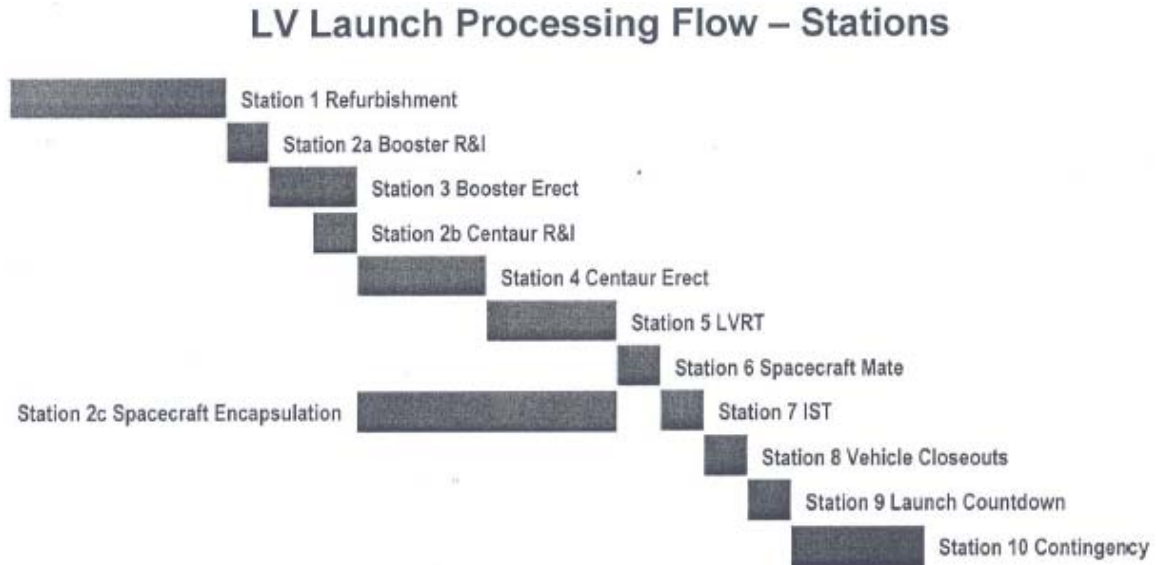
**Figure 4: Delta IV (Delta Payload Planner's Guide, 2000:1-3)**

variations associated with manual operations (Centore, 2005:12). Finally, both Evolved Expendable Launch Vehicles employ common vehicle and payload interfaces to standardize vehicle processing amongst the different vehicle variations (Evolved Expendable Launch Vehicle, 2005:par. 1,3; GAO, 2004:2). An example of a prelaunch operations schedule for Delta IV is given in Figure 5.



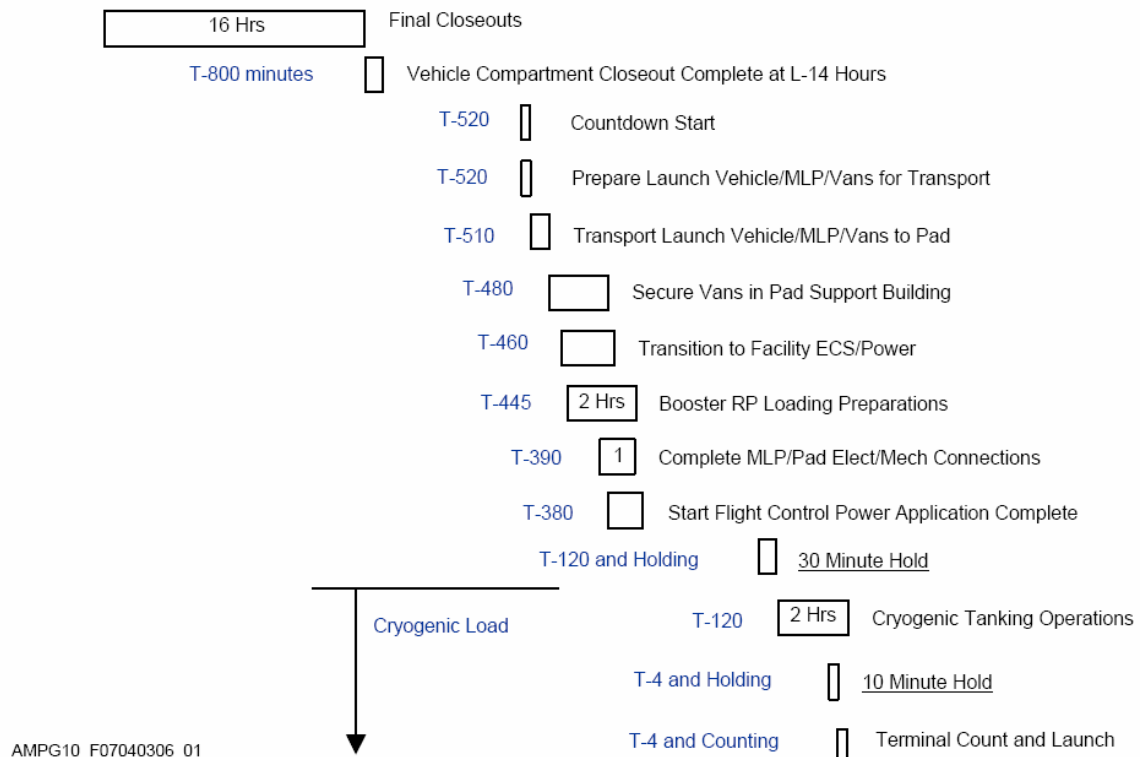
**Figure 5: Delta IV Medium Launch Vehicle CCAFS Projected Processing Timeline, Time in Days (Delta IV Payload Planner's Guide, 2000:6-34)**

Figure 5 displays a typical prelaunch operations schedule for the Delta IV Medium Launch Vehicle. Other Delta IV variations display similar schedules and possess a prelaunch operations timeline between 17 and 24 days, depending on the specific vehicle (Delta IV Payload Planners Guide, 2000:6-34-6-40). Figures 6 and 7 depict a typical prelaunch operations flow for Atlas V. Figure 6 depicts activities prior to launch day, and Figure 7 shows launch day activities.



**Figure 6: Atlas V Processing Activities Prior to Launch Day (Centore, 2005:13)**

**Figure 7.4.3.6-1: Atlas V Launch Countdown - CCAFS**



**Figure 7: Atlas V Launch Day Activities (Centore, 2005:15)**

The Atlas V launch team has the capability to complete the entire prelaunch operations schedule, from the start of integration to launch, in 18 to 26 “M days,” depending on the specific vehicle and launch requirements. One M day is equal to two eight hour shifts. This is a significant improvement over the older Atlas models (Atlas II and Atlas III), which took between 37 and 50 M days to complete the same activities (Centore, 2005:12).

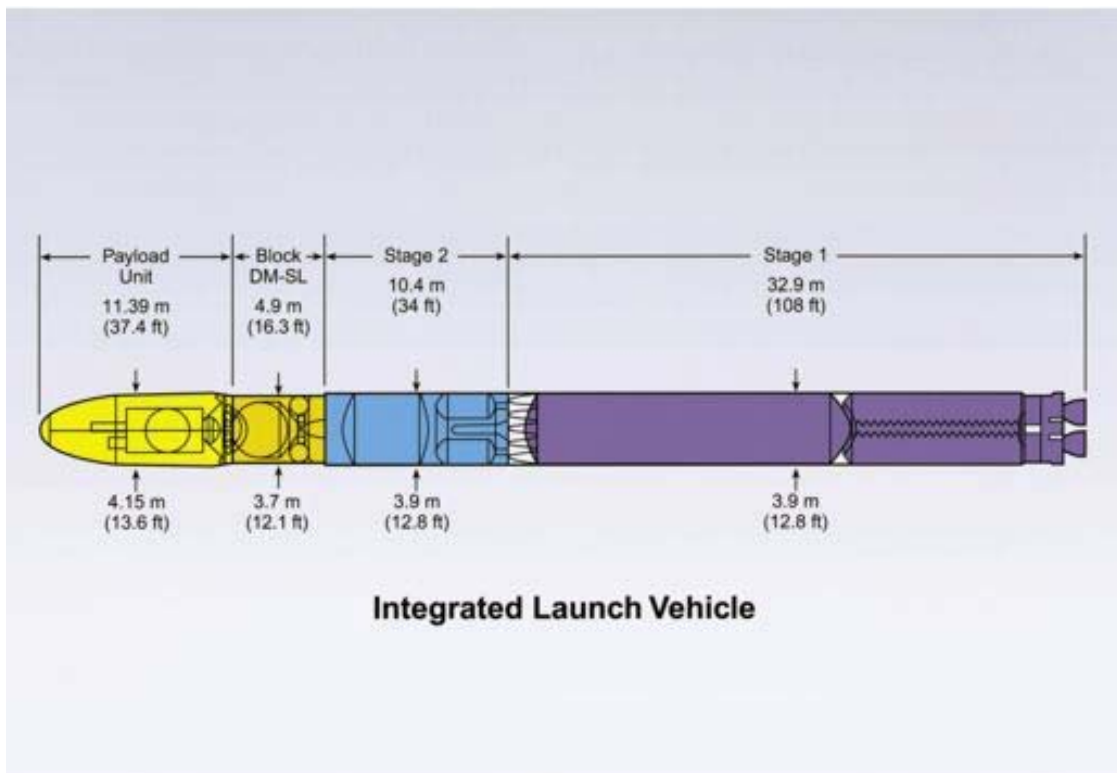
It must be noted that the schedules shown in Figures 5-7 are based upon non-continuous operations—the technicians performing these tasks do not work around the clock. The schedules could be shortened somewhat with 24 hour, seven-day-work-week schedules. Such continuous operations may be more representative of future RMLV operations. But even with this caveat, the schedules clearly demonstrate that the Air Force must improve prelaunch operations significantly over current Evolved Expendable Launch Vehicle prelaunch operations. Evolved Expendable Launch Vehicle improvements have undoubtedly shortened the amount of time it takes for launch vehicle integration and pad operations, but they still fall short of the desired ability to complete these activities within a matter of hours.

Another ELV that warrants discussion is the Zenit 3SL. The Zenit 3SL is a Russian-designed commercial launch vehicle that is marketed by Sea Launch, an international, joint business venture headed by Boeing. The first Zenit 3SL launch was in 1999, but the Zenit 3SL is based upon an older version, the Zenit 2. The Zenit 2 is still operational and is launched from the Baikonaur Cosmodrome in Kazakhstan. The Zenit 3SL is launched from a floating launch platform at a site on the equator in the Pacific Ocean (Isakowitz et al, 2004:540-543). Zenit is known as somewhat of a benchmark for

efficient and timely prelaunch operations. The Russians designed the vehicle and supporting infrastructure to facilitate quick vehicle and payload integration and rapid pad operations. Zenit 2 undergoes vehicle and payload integration in a horizontal orientation. The Zenit 2 and associated payload is fully integrated and tested only 91 hours after the individual Zenit 2 stages arrive at the integration facility. Once the vehicle is integrated and tested, it is transferred to transporter/erector railcars for transport to the launch pad. Transport and pad operations also go quickly—launch usually occurs within 28 hours of the vehicle leaving the integration facility. Once the vehicle reaches the pad, an automated process completely controls the erecting of the vehicle and umbilical connections. As soon as the vehicle lifts off the pad, the vehicle supports and autocouplers retract into the pad where they are shielded from the exhaust. The Russians designed the vehicles and pad so that a second launch could take place only 90 minutes after a previous launch (Isakowitz et al, 2004:557).

The Zenit 3SL reserved many of the processing capabilities of the Zenit 2. The main differences are that integration takes place on an Assembly and Command Ship and that launch occurs at sea from a floating launch platform. Stage and payload integration and associated testing takes place in a horizontal orientation on the main deck of the Assembly and Command Ship at the Sea Launch Home Port in Long Beach, CA. After integration is complete, the vehicle is lifted via crane onto the floating launch platform. Once the vehicle is safely onboard, the launch platform starts its journey to the launch site at the equator, which normally takes 10 to 12 days. Launch takes place three days after the launch platform reaches the launch site. While in transport, the vehicle resides in a covered hangar on the launch platform. On launch day, the vehicle is rolled to

launch position and erected and fueled with the same automated system used for the Zenit 2 (Isakowitz et al, 2004:558). Much can be learned from Zenit prelaunch operations methods and possibly applied to future Air Force RMLV operations. Figure 8 shows an integrated Zenit 3SL, and Figure 9 illustrates a Zenit 3SL erected on the launch platform.



**Figure 8: Zenit 3SL (Sea Launch Image Gallery)**

Some may question the credibility of comparing prelaunch operations for ELVs to prelaunch operations for RMLVs. While RMLVs differ from ELVs in a variety of ways, they also share many similarities, especially concerning their respective prelaunch operations activities. Just like ELVs, RMLVs will undergo some sort of stage mating process, at least in the foreseeable future, since single stage to orbit designs appear unfeasible with current technology. Payload integration will more likely mirror the Evolved Expendable Launch Vehicle concept rather than the shuttle concept.



**Figure 9: Zenit 3SL atop the Floating Launch Platform (Sea Launch Image Gallery)**

Evolved Expendable Launch Vehicles use encapsulated payloads mated with common payload interfaces, but shuttle payloads are integrated inside payload bay doors and often necessitate significant shuttle modifications to accommodate specific payloads. These modifications add a significant amount of time to shuttle regeneration operations (McCleskey, 2005:41). If the Air Force's RMLV design necessitates a vertical launch, it must be situated vertically on the launch pad, just like an ELV. Finally, most launch pad activities will be similar, since RMLVs will require umbilical (electrical, communications, and propellant lines) and mechanical connections at the pad along with vehicle tests similar to ELVs.

### ***RMLV Alert Ability and Prelaunch Operations***

The simulation model built for this thesis will include only integration, transport, and launch pad activities that take place after RMLV inspection and repair is complete. RMLV maintenance actions are examined via concurrent research by Pope (Pope, 2006). Until now the reader may have assumed that prelaunch activities always begin immediately upon the completion of RMLV maintenance (inspection and repair activities), but this will not always be the case. The commencement of RMLV prelaunch operations will be directly linked to a need to launch; they may not begin until just prior to a scheduled launch or until a need for launch arises. Current space launch systems fly according to a fixed launch schedule that is often determined months or years in advance, but some future RMLV operations may be more *reactive* in nature. It is likely that the Air Force will eventually keep one or more RMLVs on alert status. In fact, some authors have compared future RMLV operations to B-52 alert operations in the Cold War era (Brown, 2004b:5,7). These B-52s were “cocked” and completely ready to fly on a moment’s notice. However, an RMLV in alert status will not be able to launch as quickly as an aircraft since it would be infeasible for it to remain for extended periods on a launch pad completely fueled and ready to fly. It would likely have all or some prelaunch operations activities that it must go through before it could launch. For instance, an RMLV on alert may still be in “pieces.” Its stages and payload may be completely prepped and ready, waiting in a hangar. Once the launch order is given, these stages and payload would still need to be integrated, transported to the pad, and fueled before launch could take place. This is another reason that well-planned, efficient, and rapid prelaunch



operations are so important. In an alert scenario, the time required to launch after the launch order is given will directly depend upon the length of prelaunch operations.

### **Previous Launch Vehicle Simulation Research**

Researchers have been using simulation models to analyze launch vehicle ground operations for quite some time. However, most of these models have been built in conjunction with NASA and focus exclusively on shuttle operations or are heavily influenced by shuttle data. While these models are extremely useful for analyzing shuttle operations, this thesis attempts to build a model that goes beyond the shuttle mindset and is more useful for analyzing potential RMLV operations.

In 1982, less than a year after the shuttle began operational service, NASA initiated a modeling effort to analyze the feasibility of different launch schedule options. NASA was experiencing unexpected delays in shuttle processing and wanted a tool to help them develop a schedule that more closely reflected true regeneration capabilities. The model developed by Wilson, Vaughan, Naylor, and Voss was named the Shuttle Traffic Evaluation Model (STEM) (Wilson et al, 1982:190). STEM was an early model and suffered from lack of historical data. Although it did demonstrate that shuttle operations would take longer than originally expected, it still estimated some regeneration times as low as 28 work days, a capability that NASA was never able to reach (Wilson and others, 1982:197).

Shuttle Ops is a more recent shuttle simulation model that accurately mirrors true shuttle regeneration capability. In 1999, NASA was evaluating the feasibility of increasing the shuttle flight rate from 7 to 15 flights per year. NASA needed to know

which existing resources would be capable of handling the increased flight rate and which resources would need to be supplemented. Discrete event simulation was chosen as the tool to answer this question, due to the large number and complexity of processes involved with shuttle regeneration. Shuttle Ops was developed in Arena software through a joint effort between NASA and the University of Central Florida (Cates and others, 2002: 754). The developers collected and analyzed historical data on task completion times for shuttle ground processing activities. This data was used to fit probability distributions that were assigned to processes within the Arena model (Cates and others, 2002:759). The attempt to capture the myriad of activities that make up shuttle regeneration was a significant task and resulted in nearly 1,000 Arena program modules in the final model (Cates and others, 2002:757). The developers verified and validated the model by comparing model output to actual historical data. They knew they had a credible model when certain model outputs such as flight rate per year and time spent on pad were similar to historical data from periods that mirrored the model environment (Cates and others, 2002:760,761). Even though NASA gave up the idea of increasing the shuttle flight rate before Shuttle Ops was completed, the tool was still used successfully to model other scenarios such as mothballing a shuttle orbiter or closing shuttle facilities (Cates and others, 2002: 761).

More recently, Shuttle Ops was modified to create the Manifest Assessment Simulation Tool (MAST). MAST estimates probabilities of completing shuttle launches according to shuttle manifests (Cates, 2005:3). A shuttle manifest is a schedule that outlines starting and completion times for major shuttle activities, such as orbiter maintenance, vehicle assembly, and launch pad operations. MAST has been used to

demonstrate the low probability of achieving the planned number of shuttle launches before shuttle retirement in 2010 (Cates, 2005:25,26).

NASA engineers also developed a simulation model that can be applied to any type of launch vehicle. The Generic Simulation Environment for Modeling Future Launch Operations (GEMFLO) was developed in conjunction with NASA's Space Launch Initiative (SLI). The SLI program studied different RLV design alternatives as a replacement for the space shuttle. NASA developed GEMFLO to estimate flight rates and other capabilities for competing RLV designs (Steele and others, 2002:750). GEMFLO also runs in Arena software and utilizes a Visual Basic Graphical User Interface (GUI) (Steele and others, 2002:751). The main benefit of GEMFLO is that it is generic and can be used to analyze any RLV design without any model modification. One model can be used to evaluate all vehicle designs instead of building a separate model for each vehicle design (Steele and others, 2002:747,748). Accordingly, the model relies upon a large amount of user inputs as to the estimates of activity process times and other capabilities. GEMFLO takes user-inputted probability distributions and other information and then populates the Arena model with this data (Steele and others, 2002:751). GEMFLO provides outputs such as estimates of vehicle flight rate per year and vehicle regeneration time. The model developed for this thesis will in many ways be similar to GEMFLO. Like GEMFLO, this prelaunch operations simulation model will be generic. In other words, the same model will be used to analyze different vehicle designs and different ground operation variations. However, the model developed for this thesis differs from GEMFLO in that it breaks prelaunch operations down into more detail than is provided in GEMFLO. The purpose of the author's model is to analyze launch

operations in a military environment, where a quick response to military contingencies is necessary. Since Air Force requirements will dictate RMLV turnaround within a matter of hours, even processes that only take a matter of minutes will be important to analyze. GEMFLO, while a significant modeling accomplishment in its own right, does not break higher level processes down into the smaller processes that are required for such a time-saving analysis. This thesis seeks to break down higher level processes such as “Vehicle Integration” and “Launch Pad Operations” into their basic components so that model users can more accurately determine where time is being used.

Rooney and Hartong developed an Arena-based simulation model that estimates maintenance task completion times for RMLVs. Like GEMFLO, their model also includes a Visual Basic GUI. The user inputs vehicle design parameters, such as amount of thermal protection system tile area and other resource and job sequencing information. The model feeds these inputs into Arena and estimates total turnaround time and turnaround time for specific vehicle subsystems (Rooney and Hartong, 2004:6,7). This model does break processes down into an adequate level of detail, but processing time distributions are heavily influenced by shuttle historical data, which may or may not represent future RMLV operations. This model does not include any prelaunch operations activities (Rooney and Hartong, 2004:7).

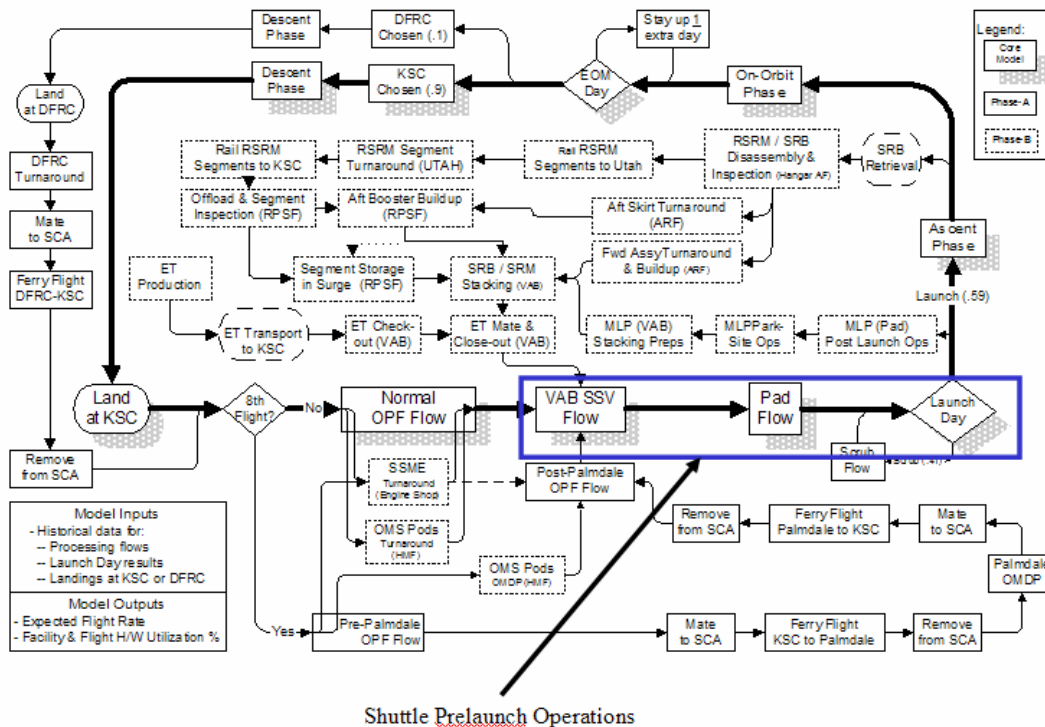
### **Prelaunch Operations Comparisons**

This section will compare predicted RMLV prelaunch operations to shuttle and ELV prelaunch operations and to similar operations for aircraft and ICBMs. This comparison is necessary for two reasons. First, since the RMLV does not yet exist, it is

difficult to describe its prelaunch operations sequence. However, insight into a predicted RMLV prelaunch operations sequence can be generated by piecing together applicable processes from aircraft, ICBMs and other launch vehicles. Analyzing existing systems in this way allowed the author to answer the first and second investigative questions in Chapter 1; he constructed a RMLV prelaunch operations sequence by picking and choosing appropriate activities from existing systems. His activity choices were guided by the literature and the Delphi study discussed in the “Validation” section in Chapter 3. Second, such a comparison between RMLVs and aircraft will give the reader an appreciation for the unique challenges involved with RMLV prelaunch operations. The RMLV is first compared to the shuttle, then to aircraft, then to ELVs, and finally to ICBMs.

### ***Shuttle Comparison***

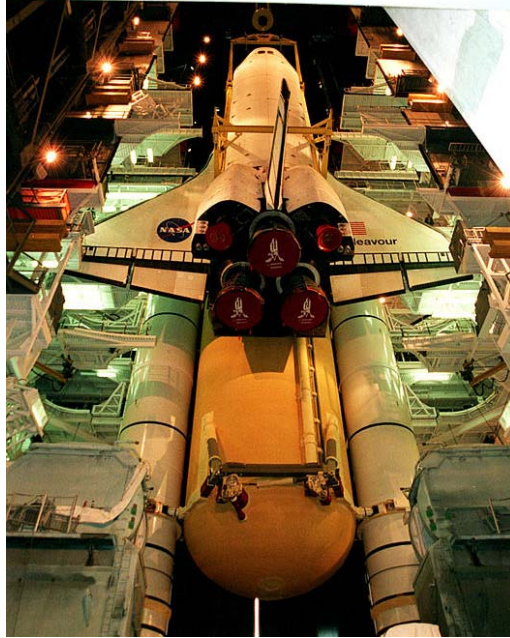
Shuttle prelaunch operations include Vehicle Assembly Building operations, transport to the launch pad, and launch pad operations. A graphical representation of the entire shuttle regeneration process is given in Figure 10. The boxed-off portion in Figure 10 includes shuttle prelaunch operations. There are three major Flight Hardware Elements that make up a space shuttle vehicle: the orbiter, the external tank, and a pair of solid rocket boosters. The solid rocket boosters are built up in the Vehicle Assembly Building and joined to the external tank upon a mobile launch platform, but for the purposes of this paper, these activities are not included as prelaunch operations, since they are considered preliminary steps to prepare Flight Hardware Elements for integration (Cates et al, 2002:755,756). Prelaunch operations for the shuttle begin once the orbiter reaches the Vehicle Assembly Building. The orbiter is attached to a large



**Figure 10: Shuttle Prelaunch Operations (Cates and others, 2002:758)**

slings and is lifted by a crane into its integration position, where it is then joined to the external tank and solid rocket boosters. This process is depicted in Figure 11. At this point, a fully integrated space shuttle vehicle is sitting in vertical position atop the mobile launch platform.

Once integration checks are complete, a crawler/transporter is used to move the mobile launch platform with the space shuttle vehicle to the launch pad (Cates, 2003:108-112). Figure 12 shows the space shuttle vehicle being transported to the launch pad. The crawler/transporter delivers and secures the mobile launch platform to the launch pad and then drives away. Many activities take place at the launch pad before launch can occur, and only the major activities are listed here. Umbilical connections are secured and pad



**Figure 11: Orbiter being mated to the rest of the space shuttle vehicle (Cates, 2003:111)**



**Figure 12: Space shuttle vehicle being transported to the launch pad (Cates, 2003:112)**

validation takes place to ensure that the mobile launch platform and space shuttle vehicle are properly connected to the launch pad (Cates, 2003:125). Payloads that require vertical integration are integrated on the pad. Certain hazardous operations such as the loading of toxic hypergolic fuels and installing of ordnance are held off until the last moment they can occur to minimize risk to personnel. Launch day events include crew ingress and loading of the main engine propellants, liquid hydrogen and liquid oxygen, into the external tank (Cates, 2003:115). Figure 13 depicts payload installation operations at the launch pad.



**Figure 13: Payload installation operations at the launch pad (Cates, 2003:126)**

RMLV prelaunch operations will share many similarities with shuttle prelaunch operations, especially when it comes to the major functions that must occur. An RMLV will require an integration phase, like the shuttle. The various RMLV stages must be assembled into a complete RMLV, and its payload must be attached. However, RMLV integration will ideally be much more streamlined than shuttle integration. The shuttle is integrated in a facility that was not originally intended for shuttle integration; this requires delicate orbiter maneuvering between building structural supports and adds time



to the integration process (Vehicle Assembly Building, 1999). In addition, depending on vehicle design, the RMLV integration process could occur in a horizontal orientation, but all shuttle integration takes place in a vertical orientation. There are benefits to horizontal integration; vehicle access is easier as most tasks can be conducted at ground level. Vehicle handling can be less complicated since stages and payload do not have to be hoisted by crane into position. Many safety concerns are also avoided with horizontal integration, since technicians do not have to perform integration tasks many stories above ground level. Finally, a fully assembled RMLV will likely look very different from a space shuttle vehicle and will thus necessitate different integration equipment and activities. Some notional RMLV designs are depicted in Figures 14 and 15. Since RMLV design is not finalized, many of the finer points of RMLV integration are yet to be determined. But the comparison to the shuttle offers helpful advice: RMLV designers should design the RMLV with a simple, quick integration process in mind. Any design alternatives that necessitate complicated and time-consuming integration procedures will add unwanted time to the regeneration process.



**Figure 14: Notional RMLV Design (Rooney and Hartong, 2004:8)**



**Figure 15: Notional RMLV Design (“Hybrid Launch Vehicle,” 2005:1)**

RMLV launch pad operations will likely differ from shuttle launch pad operations in several key areas. First, the RMLV will be an unmanned vehicle, at least for the foreseeable future, but all shuttle missions are manned. This means crew ingress and related safety procedures will not be a concern for RMLV launch pad operations. Second, as with integration operations, RMLV launch pad operations will ideally be much more streamlined than shuttle launch pad operations. With the shuttle, a myriad of umbilical connections must be attached and verified manually; RMLV connections will hopefully be fewer and attached automatically, as is the case with the Zenit ELV. Finally, propellant choice will likely be different for the RMLV. The shuttle uses liquid hydrogen as a fuel and liquid oxygen as an oxidizer, but RMLV propellant combinations will likely be either RP-1 and liquid oxygen or methane and liquid oxygen (Dornheim, 2005:34). RP-1 is a kerosene-based fuel and is non-cryogenic, which makes its storing and handling much easier and safer than cryogenic fuels, like liquid hydrogen. RP-1 fueling can actually be done in parallel with other operations, but cryogenic fueling cannot. Methane is a cryogenic fuel, but it may be used due to performance and cost

benefits (Brown, 2004b:23). Hypergolic fuels like hydrazine, which are used on the shuttle, will also ideally be avoided for RMLVs since they are toxic and dangerous to handle (Dornheim, 2005:34). Hypergolic fuels can lengthen regeneration time since other activities cannot be done in parallel with hypergolic fuel loading.

### *Aircraft Comparison*

A common phrase being used within the RMLV community is “aircraft-like operations” (Dornheim, 2005:34). The hope is that the RMLV sortie rate will approach sortie rates experienced by aircraft. As already discussed, significant improvements in RMLV regeneration time over existing systems must be made before this will happen. Since aircraft-like regeneration time is the goal for RMLVs, it will be helpful to compare RMLV prelaunch operations to similar aircraft regeneration activities to see which, if any, RMLV prelaunch activities will prevent RMLVs from becoming truly “aircraft-like.”

In most cases, the military aircraft regeneration process is designed to happen very quickly. Unlike the space shuttle, aircraft do not undergo extensive scheduled maintenance between every sortie. Scheduled replacement of limited-life components and major inspections and overhauls occur during aircraft “downtime,” either at the end of the flying day or during periodic inspection time periods specifically designated for that purpose. Of course specific requirements vary from aircraft to aircraft, but the major events that take place between aircraft sorties include safing, inspection, repair, servicing, and “payload” installation. Aircraft safing includes those events required to make the aircraft safe for maintenance, such as installation of landing gear safety pins and ordnance safety pins and grounding the aircraft to eliminate static electricity dangers.

Aircraft inspection includes visual examination of aircraft components and structure to look for damage or signs of impending failure. Inspection also includes operation of certain aircraft systems to make sure they are working properly. Repair is not a certain requirement between every sortie. Repairs only take place if the aircraft malfunctioned during flight or if a fault is discovered during inspection. Servicing includes the loading of certain fluids and gasses such as jet fuel, liquid oxygen, hydraulic fluid, gaseous oxygen, and gaseous nitrogen. The type of payload installation required depends upon the aircraft's mission. The payload could be cargo, personnel, munitions, or sensors.

Of these five events just described, servicing and payload installation are the ones that parallel RMLV prelaunch operations. However, there are several key differences between aircraft servicing and payload installation and RMLV servicing and payload installation. These differences make RMLV prelaunch operations more complicated and thus more time-consuming. First, RMLVs require both fuel and oxidizer for engine operation. The oxidizer will almost certainly be liquid oxygen (Brown, 2004b:24), which is cryogenic, and the fuel may or may not be cryogenic. In contrast, aircraft engines operate on kerosene-based jet fuel, which is similar to the RP-1 rocket fuel discussed earlier. While there are safety guidelines for handling jet fuel, they are not nearly as stringent as those required for cryogenic loading operations. No personnel are allowed near a launch vehicle while cryogenic propellant loading is taking place, which precludes the ability for technicians to perform parallel tasks. Additionally, cryogenic fuels cannot be loaded far in advance of launch because too much of the fuel would “boil-off” and be lost. Substances like oxygen, hydrogen, and methane exist as gasses at normal ambient temperatures, so keeping them in a liquid state is difficult. This is why cryogenic loading

operations always take place just prior to launch. Aircraft are not constrained by this requirement, which gives them more flexibility in the timing of fueling operations. An aircraft can be refueled days before its next sortie if necessary.

The sheer amount of fuel needed for an RMLV launch also presents a significant difference from aircraft. Rocket engines must consume an enormous amount of fuel to propel a launch vehicle into orbit. For instance, the shuttle's external tank holds 528,616 gallons of propellant (fuel and oxidizer combined) (Cates, 2003:95). Atlas V used 191,365.2 gallons of propellant (fuel and oxidizer combined), for the Pluto New Horizons launch in January of 2006 (Andrews, 2006). In contrast, even a heavy bomber aircraft like the B-2 holds only 22,000 gallons of fuel at its max capacity (O'Malley, 2006). RMLVs will require significantly more fuel than aircraft, and this fact in itself adds to the length of propellant loading time.

Attaching a payload to a launch vehicle is much more complicated than loading bombs on aircraft. Launch vehicle payloads often come with special handling requirements; some payloads must constantly remain in a vertical position, and most payloads require a constant supply of conditioned air. After payload attachment, extensive interface tests are often required to verify normal spacecraft operation and proper payload to launch vehicle connection. Aircraft payloads, whether they are bombs, people, or cargo, are normally not constrained by these requirements, and uploading them can thus occur more quickly.

In addition to differences between aircraft and RMLV servicing and payload operations, RMLV prelaunch operations include activities not required for aircraft. First, prior to prelaunch operations, RMLV stages are processed separately. These stages must

be joined, or “integrated,” to form the entire launch vehicle. This is not a consideration for aircraft, since the entire aircraft lands in one piece and stays in one piece throughout the regeneration process. Second, RMLVs require more ground handling than aircraft since RMLVs must be placed on the launch pad by means external to the vehicle itself. Individual RMLV stages must be transported to the integration facility, and then the entire vehicle must be transported to the launch pad. Assuming a vertical takeoff with horizontal landing configuration, the RMLV must also be erected to a vertical position on the launch pad if it was integrated horizontally. Even though aircraft must sometimes be towed during ground operations, they do not require such extensive handling. An aircraft can taxi via its own engine power to the runway and then take off.

The above list of differences between RMLVs and aircraft is not exclusive, but it does demonstrate that RMLV prelaunch operations are much more challenging than similar aircraft activities. These additional challenges add time to RMLV prelaunch operations and indicate that these operations may never be truly “aircraft-like.”

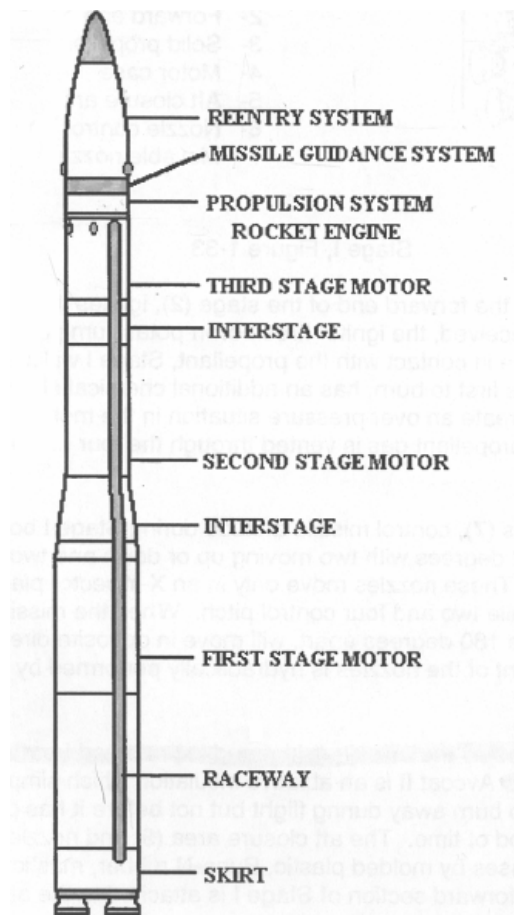
### ***ELV Comparison***

RMLV and ELV prelaunch operations will likely be similar, at least at the macro level. However, most of the differences between RMLVs and the shuttle also apply. Specific integration equipment and procedures will differ depending on vehicle design. While in a vertical orientation, ELV stages are stacked on top of each other, but RMLV stages may be joined differently. In a vertical orientation, most RMLV notional designs depict RMLV stages stacked side by side instead of on top of each other (see Figures 14 and 15).

### ***ICBM Comparison***

ICBMs and RMLVs perform different missions, but they possess enough similarities to make their comparison worthwhile. RMLVs will put payloads into orbit, but ICBMs release their payloads without putting them into orbit. However, both vehicles are propelled by powerful rocket engines and can reach comparable top speeds. ICBMs also require integration of stages and payload like an RMLV.

The Minuteman III is the best example of current ICBM operations as all other ICBMs have been retired. It is depicted in Figure 16.



**Figure 16: Minuteman III (ICBM Familiarization, 2001:60)**

The Minuteman III has the ability to deliver multiple warheads to independent targets. It consists of three solid propellant stages collectively termed the “downstage,” a post boost control system for payload maneuvering after downstage separation, and a reentry system which houses the payload, or warheads (ICBM Familiarization, 2001:60). These components must be integrated to form a fully functional ICBM.

Minuteman III assembly takes place in a missile silo in a vertical orientation. The downstage is the first piece lowered into the silo by a Transporter Erector (see Figure 17).



**Figure 17: Transporter Erector (LGM 30 Minuteman III)**



The downstage is always lowered as a single piece; even though it is made of three stages, these stages are never separated except at depot (Pope, 2005). Once the downstage is secured, the post boost control system is lowered and mated with the downstage. Finally, the reentry system is lowered and mated to the post boost control system (see Figure 18). Mating connections between the ICBM components are relatively simple, consisting of a row of screws around the circumference of the missile and several cannon plugs (Pope, 2005). Only two silo to missile umbilical connections need to be made (Pope, 2005).



**Figure 18: Minuteman III Reentry System Install (LGM 30 Minuteman III)**

Many of the handling and processing characteristics of the Minuteman III stem from ICBM alert requirements. Since the ICBM mission necessitates the maximum amount of missiles in operational status, eliminating unnecessary maintenance and handling time is important. Several Minuteman III features offer suggestions for possible time-saving measures for RMLV prelaunch operations. First, since the downstage remains in one piece, each of the three solid propellant stages do not need to be integrated

separately. Likewise, any RMLV assembly that can be done ahead of time should be. For example, an RMLV on alert should be as fully assembled or as “pre-integrated” as possible to decrease the amount of assembly required after a launch command is given. An RMLV’s lower and upper stage(s) could be pre-integrated with only the payload left to attach. Or perhaps the payload and upper stage(s) may be pre-integrated, with only the lower stage left to attach.

The identical nature of each Minuteman III reentry system provides another advantage. ICBM payload to missile interfaces are standardized (Pope, 2005); technicians do not have to configure each missile to adapt to different payload connections since each payload attaches in the same way. This facilitates timely payload integration since it negates specialized and time-consuming missile reconfigurations and allows missile maintainers to become very proficient at one set of payload integration procedures. RMLVs will be able to avoid extensive shuttle-like payload modifications to the extent that future RMLV payload interfaces can be standardized like ICBM payload interfaces. This may be more difficult for RMLVs since their payloads will vary greatly from mission to mission, but payload interfaces must be as simple and as standardized as possible to minimize payload integration time.

ICBM integration operations illustrate the disadvantages of vertical integration. Much of the time spent on ICBM integration is directly related to the requirements associated with working on a missile in a vertical orientation. Access to a missile standing upright in a silo is difficult. Several silo access doors allow access to limited areas of the missile. The only way to access most missile areas is by using an elevator workcage, a two-person structure suspended by cable from a winch positioned at the silo

opening (see Figure 19). A work environment like this increases risk of injury and adds time to the overall operation. Workers must don safety harnesses and secure themselves to lanyards to keep them from falling. In addition, all tools used by the workers must be attached to lanyards to prevent the tools from falling and damaging the missile (ICBM Familiarization, 2001:77-79). An elevator workcage will likely not be necessary for a vertically integrated RMLV, since launch vehicle vertical integration facilities utilize extensive scaffolding assembled around the vehicle. However, the other disadvantages of vertical integration cannot be avoided unless the vehicle is integrated horizontally.



**Figure 19: Technicians Working from Elevator Workcage (LGM 30 Minuteman III)**

There are several differences between ICBMs and RMLVs that may never allow RMLVs to be as responsive as ICBMs. First, since the Minuteman III uses solid propellants for its downstage, propellant loading immediately prior to launch is not necessary. Earlier ICBM versions used liquid propellants, but this concept was abandoned since propellant loading could not take place until immediately prior to launch (Neufeld, 1990:203, 233, 237). Liquid propellant loading lengthened the response time for ICBMs. Since RMLVs will almost certainly use liquid propellants (Brown,

2004b:21-25), propellant loading immediately prior to launch will always be a necessity. Second, an ICBM on alert always has its payload attached, but this may not be the case for RMLVs. In an RMLV alert scenario, the type of payload required may not be known in advance, which means that payload integration must take place prior to launch. This will lengthen RMLV response time, as previously discussed. Finally, an ICBM remains on alert in its silo, which is its launch site. RMLVs will probably not remain on alert on a launch pad since this would tie up the launch pad and would require the construction of additional launch pads for scheduled launches. Since launch pad construction and associated support equipment is so expensive, an alert RMLV would likely remain off the launch pad and be moved to the launch pad at launch time. The time required for transport to the launch pad will also lengthen RMLV response.

## **Summary**

This chapter provided the reader with a background of the research problem and discussed the significance of the problem. The first section covered future spacelift objectives for both NASA and the Air Force. The next section discussed the unresponsive nature of current space launch systems. This was followed by an overview of the existing research pertaining to launch vehicle ground operations simulation. The chapter concluded with a comparison of RMLV prelaunch operations to shuttle, aircraft, ELV, and ICBM prelaunch operations. The next chapter will outline the methodology used in this research.

### **III. Methodology**

#### **Introduction**

This chapter explains the methodology used to develop a RMLV prelaunch operations simulation model. The first section will reiterate the need for such a model and explain how such a simulation model will be useful for decision makers. The second section will describe the specific steps undertaken to develop this model.

#### **Applicability of Discrete Event Simulation to the Research Problem**

The purpose of this thesis is to create a tool that will help Air Force decision makers analyze different RMLV design and operational concept alternatives. The tool created will be a discrete event simulation model that will evaluate how different RMLV configurations will affect prelaunch operations flow. This particular research effort is complementary to concurrent maintenance modeling efforts by Pope (Pope, 2006) for the Air Force Research Laboratory Air Vehicles Directorate (AFRL/VA). The simulation model developed for this thesis only covers prelaunch operations, but it was joined with Pope's model to create a larger model that covers both maintenance and prelaunch operations. The combined maintenance and prelaunch operations model is called the Maintenance, Integration, and Launch Pad Operations Simulation and Test (MILePOST).

The Air Force is in the early stages of its RMLV program. This early stage is especially critical for future program success since leaders are now making decisions about RMLV design and ground operations that will determine how quickly RMLV regeneration can take place. Researchers and leaders need to know how their decisions

will affect the amount of time and resources that will be needed to maintain a fleet of RMLVs. Discrete event simulation is an appropriate and useful tool for such a problem.

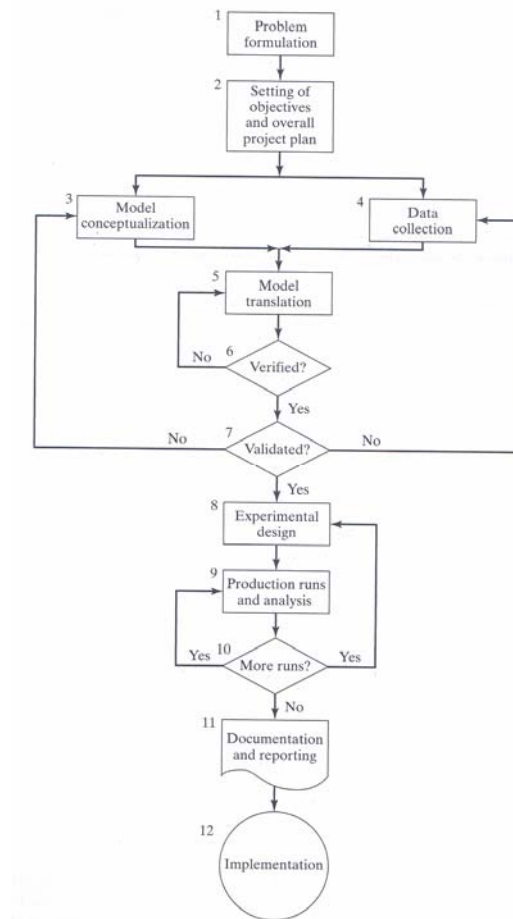
“A simulation is the imitation of the operation of a real-world process or system over time” (Banks and others, 2005:3). More specifically, discrete event simulation is “the modeling of systems in which the state variable changes only at a discrete set of points in time” (Banks and others, 2005:13). If the system being simulated is RMLV operations, the system will be a discrete system; the state of the system will change at discrete points in time, not continuously. For instance, the state of RMLV operations changes when an RMLV launches, and this change happens at a discrete point in time.

Simulation is not an appropriate methodology for every research problem, but it will be especially useful for analyzing RMLV operations for the following reasons. First, simulation is a useful tool for situations in which direct experimentation with the real-world system is not feasible. This is certainly the case with RMLV operations as the RMLV does not yet exist. Second, simulation is fitting for analyzing extremely complex systems. If a system or process is simple, with few variables and few interactions amongst those variables, then a problem or question associated with that system can often be solved by common sense or direct mathematical computations. RMLV operations however, represent very complex systems due to the myriad of resources and processes involved. In such a situation, a simulation model developed and run on a computer is the only feasible way to analyze the system. Third, simulation models allow users to adjust system flow and system inputs for the purpose of seeing how the overall system will react to such adjustments. An RMLV simulation model would thus allow users to change RMLV design variables and different ground processing options to see how these

changes would affect RMLV regeneration time. Finally, the information gathered and knowledge gained in building a simulation model are invaluable assets in themselves. Simulation model builders often gain fresh insights and more in-depth knowledge about the system under investigation. Such a byproduct will be useful to those involved with developing the Air Force's concept of RMLV operations.

### **Model Development**

Banks et al. describe a 12-step process to building a simulation model (see Figure 20) (Banks and others, 2005:15). These steps are meant to apply to any model building effort and provide a framework for describing the steps undertaken to develop the model for this thesis. The upper half of the schematic (Steps 1-7) in Figure 20 depicts the model building and validation and verification phase. Steps 1-7 represent the effort undertaken to build, validate, and verify a model for AFRL use. The lower half of the schematic (steps 8 through 12) refers to the actual use of a model to analyze a system and make decisions about that system. Since analyzing RMLV maintenance and prelaunch operations together is of more value than simply analyzing prelaunch operations alone, Steps 8 through 12 were applied to MILEPOST. The 12-step modeling process as it applies to this thesis is described in the following section.



**Figure 20: The 12 Steps in a Simulation Study (Banks and others, 2005:15)**

## **The 12-Step Modeling Process**

### ***Step 1: Problem Formulation***

In this first step a clear understanding of the research problem is formulated to guide the entire modeling effort (Banks and others, 2005:14). The Air Force is in the early phases of its RMLV program and needs information on how RMLV design and different processing flow options will affect RMLV operations.



### ***Step 2: Setting of Objectives and Overall Project Plan***

The second step involves setting goals for model development and use (Banks and others, 2005, 14). The goal of this research is to provide AFRL/VA with a tool it can use to analyze RMLV prelaunch operations.

### ***Step 3: Model Conceptualization***

This is perhaps one of the most critical steps of model formulation. It is at this point that the underlying framework or the overall “flow” of the model is developed. Model conceptualization involves “an ability to abstract the essential features of a problem, to select and modify basic assumptions that characterize the system, and then to enrich and elaborate the model until a useful approximation results” (Banks and others, 2005:14). Model-building is an iterative procedure; a modeler first develops a simple representation of the system under consideration and then builds upon and refines that simple representation until it suitably captures the real-world workings of the system. The author built his model in this way by analyzing prelaunch operations sequences for existing launch systems and combining this knowledge with best estimates of what a RMLV prelaunch operations sequence will include. The author started with a simple conceptual network of basic RMLV prelaunch operations events and then added more detail to that network as more knowledge was gained.

### ***Step 4: Data Collection***

Traditionally this step refers to collecting historical data on how the system has performed or functioned over time. The modeler fits probability distributions to this data and then uses those distributions within the simulation model (Banks and others, 2005, 16). For instance, Cates et al. describe the collection of data on shuttle solid rocket

booster stacking completion times for their Shuttle Ops model. A distribution was fit to this data, and the distribution was put into Shuttle Ops. Each time the model simulated a solid rocket booster stacking operation, it would “pull” random variables from that distribution (Cates and others, 2002:758-760). This research did not involve this type of data collection since there is no such RMLV data to collect. The author did, however, collect a large amount of data on prelaunch operations flows for existing launch systems. As discussed above, these systems’ prelaunch operations flows were analyzed and compared to gain an understanding of what a RMLV prelaunch operations flow will look like. By combining this data with the few estimated RMLV prelaunch operations flows that already existed in the literature, a credible model of RMLV prelaunch operations was developed. The prelaunch operations flows for the following space launch systems were analyzed for this research: shuttle, Atlas V, Delta IV, and Zenit 3SL. The shuttle was chosen because it is the only operational orbital RLV in the world. Atlas V and Delta IV were chosen because they are recent additions to the U.S. launch vehicle fleet and represent the most advanced concept of prelaunch operations. The Zenit 3SL was chosen because it was originally designed for quick prelaunch operations. In addition, as discussed in chapter two, RMLV prelaunch operations were also compared to similar operations for aircraft and ICBMs to see how activities pertaining to these systems may apply to RMLV prelaunch operations.

In addition, the author collected preliminary RMLV prelaunch operations activity duration estimates that were used to populate and run the model. Since no real-world RMLV historical data exists, duration estimates were obtained from similar activities performed on other launch vehicles, aircraft, and ICBMs. For example, the estimated

duration for the model process entitled “Leak Check Propellant Umbilicals” was based upon the similar Atlas V process, which takes about five minutes (Centore, 2005). The author gathered estimates for each activity in the model and then built a triangular distribution around each estimate. Using the triangular distribution allows for variability in model output in contrast to constant values, which produce only deterministic output. It also mimics the stochastic nature of the real world by producing process times from a distribution characterized by a minimum, most likely, and maximum value. Each estimated process duration obtained from other real-world systems became the process’s most likely value. The minimum value was obtained by subtracting 10 percent from the most likely value, and the maximum value was obtained by adding 40 percent to the most likely value. For instance, in the example above, the estimated value for the duration of an umbilical connection leak check is five minutes. This means that the most likely value used in the model for an umbilical connection leak check is 5 minutes, while the minimum and maximum values are 4.5 minutes and 7 minutes, respectively. The author chose the 10 percent and 40 percent figures based upon his own aircraft maintenance experience, which confirms that it is more likely for a ground processing task to take a longer amount of time than its most likely duration than it is for the task to take a shorter amount of time. The final model required the formulation of 39 triangular distributions. See Appendix C for the complete list.

#### ***Step 5: Model Translation***

This step refers to translating the conceptual model into a computer model. In this step, the model developer builds the computer code required by the simulation

software of choice (Banks and others, 2005, 16). This model was built in Arena computer simulation software.

### ***Step 6: Verification***

Verification is “the process of determining that a *model* and its resultant *simulation* ... accurately represent both what is required and what the [model] developer says will be built ... in accordance with those requirements” (Defense Modeling and Simulation Office, 1996: 1-5). A verified model will allow entities to flow through the model network in a logical fashion, as the model developer intended. Additionally, statistical output from a verified model will respond predictably to input changes (Banks and others, 2005:354,356).

Verification for this research involved a series of tests that assessed the logical flow of entities through the model. The model controls entity flow via a series of decision nodes. Each decision node can be described as a “switch” that directs entities down one of two or more subsequent paths. The verification tests ensured that the decision nodes were working correctly. The author used a dynamic verification technique to test the decision nodes. The Department of Defense *Verification, Validation, and Accreditation Recommended Practices Guide* defines a dynamic technique as a technique that requires “model execution” (Defense Modeling and Simulation Office, 1996:4-12). In other words, a dynamic test is carried out by actually running the model and then observing its behavior. This is in contrast to a static technique, which assesses model design apart from model execution (Defense Modeling and Simulation Office, 1996:4-7).

The specific dynamic verification technique used to verify this model was the Assertion Checking method. Assertion Checking is defined as “a verification technique that checks what is happening against what the modeler assumes is happening to guard against potential errors” (Defense Modeling and Simulation Office, 1996:4-13). The modeler assumption in this case is that the model’s decision nodes control entity flow through the model as the author intended. To test this assumption, the author placed record modules after each decision node within the model. These record modules kept track of how each decision node controlled entity flow during model execution. The Assertion Checking tests compared expected record module values with actual record module values after each simulation run. A set of matching expected and actual record module values confirmed that the decision nodes were controlling entity flow properly. The Assertion Checking results are discussed in detail in Chapter 4.

### ***Step 7: Validation***

The model validation process “ensures that a simulation conforms to a specified level of accuracy when its outputs are compared to some aspect of the real world” (Defense Modeling and Simulation Office, 1996:1-5). In other words, a validated model is a model that accurately imitates the real world system under investigation. Model validation is commonly performed by comparing model output data to real world data. For instance, the Shuttle Ops model was validated by comparing model estimates of annual launch rates to actual historical launch rate data and showing that the rates were similar (Cates and others, 2002:760-761). However, since no historical RMLV data exists, the model was validated via the Delphi method, which the Department of Defense *Verification, Validation, and Accreditation Recommended Practices Guide* categorizes as

an “informal validation technique.” Informal validation techniques “rely heavily on human reasoning and subjectivity without stringent mathematical formalism” (Defense Modeling and Simulation Office, 1996:4-1). The fact that informal techniques do not rely upon stringent mathematical analysis does not render them ineffective or inferior. On the contrary, informal techniques such as the one utilized for this thesis, when applied with the proper structure and guidelines, are considered acceptable and very useful for model validation (Defense Modeling and Simulation Office, 1996:1-8).

The Delphi method may be broadly defined as “an exercise in group communication among a panel of geographically dispersed experts” (The Delphi Method:par. 6). It often used to facilitate group decision making or to elicit expert opinion from a group of people on a certain topic. The Delphi method is especially useful when members of the group are physically distant or when the dynamics of a group meeting would stifle free thinking and open contribution from all members. The Delphi method is usually characterized by two or more rounds of questionnaires that are sent to the Delphi participants. Each participant is free to answer the questionnaire on his or her own time (within the time constraints of the study). Once each participant responds, a moderator distributes all responses to each group member. Individual responses and opinions usually remain anonymous. Once the group members review the first round of collected responses, they submit a second set of responses. Participants may or may not change their opinions based upon responses collected in the first round. Some Delphi studies continue to elicit responses until the group members come to some sort of consensus, but this is not always the aim of the study (Turoff and Hiltz:2-7).

The Delphi panel that participated in the review of this model was made up of 15 members from various organizations within the Air Force and NASA. Member names will not be listed here to protect the participants' anonymity, but the panel was made up of well-qualified individuals who possessed a wealth of knowledge concerning launch vehicle ground operations. Most of the participants came from AFRL/VA, but there was also representation from Aeronautical Systems Center, NASA, and Air Force Space Command. Receiving input from these experts greatly aided the author in the development of his model for a variety of reasons. First, since the author had limited experience in the area of launch vehicle ground operations, the panel corrected his mistakes and filled in the gaps that were present in his limited knowledge base. Second, having such a wide variety of launch vehicle experts from several government organizations allowed the author to leverage the different experiences of the panel members. Responses from differing viewpoints facilitated a broader understanding of RMLV ground operations and allowed the author to consider the full spectrum of launch vehicle processing possibilities. Overall, the Delphi panel greatly contributed to the model's credibility. In the absence of traditional model validation procedures, the Delphi panel ensured that the model was a legitimate representation of future RMLV prelaunch operations.

The Delphi study proceeded as follows. First, a panel of experts was chosen by the author and approved by AFRL/VA. Next, a visual flow diagram of the model network was sent to the panel. The model was built in Arena computer simulation software, but the simulation model itself was not sent to the panel because limited familiarity with the software would have made the model difficult to interpret. Instead, a

simplified representation of the model for panel review was created in Microsoft Visio software. This gave panel members an easy-to-read format while not sacrificing any details that were important to the model's operation. After the panel members reviewed the flow diagram, they submitted suggestions for improvement to the author. The author then compiled the responses and made changes to the model where there was consensus amongst the group members. The author then submitted the initial responses to the entire group along with the updated model, which effectively ended the first round and started the second. A total of three rounds were completed, each round containing a submission of the most updated model to the group members and a collection of their responses. The Delphi study was terminated after three rounds because the author concluded that continuing the study would not result in any more substantial input from the panel. Three rounds allowed each panel member to say all he or she desired to say and to comment on responses from other panel members. All panel member responses from each of the three rounds are included in Appendix A.

### ***Step 8: Experimental Design***

In the experimental design step, the modeler sets up the experimental framework that will be used to compare system alternatives (Banks and others, 2005:16, 17). The main purpose of this thesis was to build, verify, and validate a simulation model that AFRL can use to evaluate RMLV design and ground processing alternatives. However, the author also set up his own experimental design that compared several different RMLV prelaunch operations processing options. The author's experimental design serves two purposes. First, it provides AFRL/VA and other Air Force decision-makers with preliminary insights into how different decisions will affect RMLV prelaunch



operations. Second, it demonstrates the model's capability for comparing other decisions in the future.

To make the preliminary insights more valuable, the experimental design was applied to MILEPOST, the combined maintenance and prelaunch operations model. Each experiment compared two different processing options to see how the options affected RMLV regeneration time. For example, an experiment could analyze how an optional time-consuming activity (say, Activity X) affected model output. The experiment would run the model with the activity included and then run the model again without the activity included. Average regeneration time from both runs would be compared using the following hypothesis test:

Ho: Average regeneration time with Activity X – Average regeneration time without Activity X = 0

Ha: Average regeneration time with Activity X – Average regeneration time without Activity X > 0

If the null hypothesis were rejected in this case, one could conclude that removing Activity X would decrease average regeneration time.

#### ***Step 9: Production Runs and Analysis***

This step involves the actual execution of the experimental design. The model is run and output is analyzed in accordance with the experimental design (Banks and others, 2005:17). Regeneration time for each experiment was compared by developing a confidence interval around the difference in average regeneration time. The results of the statistical tests used to analyze RMLV ground processing alternatives are discussed in Chapter 4.

### ***Step 10: More Runs?***

At this step the modeler determines whether more analysis is needed, and, if so, develops additional experimental designs as necessary (Banks and others, 2005:17). AFRL/VA, the recipient of this model, will carry out this step as it deems appropriate.

### ***Step 11: Documentation and Reporting***

Once a modeler has built a model and used that model to gain insight into some system or problem, he or she will usually present the findings to those who will use the findings to make some sort of decision or decisions. Documenting and reporting involves compiling those findings and presenting them in a format that is appropriate for the intended audience. For this model, the completion of this step is satisfied mainly by the writing of this thesis (Banks and others, 2005:17). However, detailed documentation was also embedded within the model code to aid follow-on modelers in the use and modification of the model.

### ***Step 12: Implementation***

The objective of most simulation studies is not the creation of the model itself; rather, the end goal of most simulations is to use the information provided by the simulation to make some sort of decision and then implement that decision in the real-world system (Banks and others, 2005:17, 18). So it is in this case; the model was created to provide Air Force decision-makers with information to guide them in their RMLV design decisions. It is the hope of this author that this modeling effort and future efforts like it will help guide the Air Force as it progresses through the RMLV acquisition phase.

## **Summary**

Chapter 3 outlined the methodology used to complete the author's research. The first section reiterated the purpose of this research and demonstrated the applicability of computer simulation to the research problem. The second section discussed the method that was used to develop the author's simulation model. The following chapter describes the simulation model and covers the results of the model verification tests and experimental design.

## **IV. Results and Analysis**

### **Introduction**

This chapter begins with a description of the model created for this thesis and then discusses the results of the model verification process. The chapter concludes with a discussion of the experimental design that was applied to the model.

### **Model Description**

The author built his model in Arena computer simulation software. The model breaks the prelaunch operations process into some detail and possesses approximately 160 Arena modules. Many of the launch vehicle simulation models discussed in the “Previous Launch Vehicle Simulation Research” section in Chapter 2 simulate a higher level of launch vehicle operations. They do this by rolling up many smaller activities to create a larger process that encompasses all of the smaller activities. This means that a process, such as launch pad operations, in one of these models may be represented by a single module and distribution, while in the author’s model, this upper-level process is broken down into the many activities that comprise it. The author did this to direct attention to the specific activities that will have to be analyzed as the Air Force contemplates RMLV design and time-saving alternatives.

In addition to creating the model itself, the author used VBA code to develop a graphical user interface that simplifies user inputs to the model. The graphical user interface is not necessary for model operation, but it was added to make the model more user-friendly. Without the graphical user interface, users would be required to input

information directly into the Arena modules, which requires a working knowledge of the Arena software. However, it is likely that not all future users of the model will be familiar with Arena software. The graphical user interface circumvents this problem by allowing the user to complete all inputs using standard, familiar user forms. The VBA code transfers the user's inputs to the Arena model and prepares the model for execution per the user's instructions. The graphical user interface also responds to user inputs itself, directing the user to the appropriate questions depending upon user choices. An example of one of the graphical user interface user forms is given in Figure 21.

Appendix B includes the entire set of graphical user interface user forms.

**(OT) Other tasks, vehicle integration facility**

This form gathers user inputs that describe final tasks within the integration facility, such as hypergolic fuel loading, ordnance install, and vehicle transport preparations.

Will hypergolic fuels be required?

OT-01 ☐ hypergolic fuels required ☐ Hypergolic fuels not required

OT-07 Prepare for vehicle transport to pad 1 Hours

Click on the appropriate location for hypergolic fueling. Hypergolic fuels can be loaded now, in the integration facility, or later, on the launch pad.

OT-02 ☐ Load hypers in integration facility ☐ Load hypers on launch pad

OT-08 Attach vehicle transporter to the vehicle 1 Hours

OT-09 Transport vehicle to launch pad 1 Hours

OT-03 Time to load hypergolic fuels 1 Hours

Will ordnance be required?

OT-04 ☐ Ordnance required ☐ Ordnance not required

Click on the appropriate location for ordnance installation. Ordnance can be installed now, in the integration facility, or later, on the launch pad.

OT-05 ☐ Install ordnance in integration facility ☐ Install ordnance on launch pad

OT-06 Time to install ordnance 1 Hours

Previous Next  
Main Help

**Figure 21: Graphical user interface user form example**

The model represents RMLV processes that will occur in the future and as such simulates a real-world system that does not yet exist. The uncertainty associated with the activities that will one day be involved in RMLV prelaunch operations requires a generic modeling approach. A generic simulation model is a model that can accurately represent more than one real-world system. A benefit of generic models is that they can be applied to many different situations, which precludes the necessity of creating a separate model for every situation. This also makes generic models especially fitting for comparing system design alternatives (Steele and others, 2002: 747, 748). The model created for this thesis is a generic model in that it can be applied to many different RMLV design alternatives. The user decides which RMLV processing options to analyze and the model adjusts to accurately represent the user-defined RMLV specifications.

The model is currently configured to allow only one entity to flow through the model per replication. Each replication then, represents one complete prelaunch operations cycle for a single entity. Each entity represents a RMLV mission, not the vehicle itself. In other words, each entity represents a requirement to launch an RMLV mission. Each mission entity seizes resources (such as a first stage, maintenance hangar, or launch pad) as it proceeds through the prelaunch operations sequence. The author is indebted to the GEMFLO (a model discussed in the “Previous Launch Vehicle Simulation Research” section in Chapter 2) developers for the mission entity idea (Steele and others, 2002: 751).

Even though the model is generic and adapts to many different RMLV processing options, it does make some basic assumptions. The model assumes a reusable first stage and expendable second stage to match the Hybrid Launch Vehicle concept (see the “Air

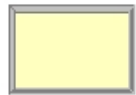
Force and NASA Future Spacelift Objectives” section in Chapter 2). It also assumes that liquid fuels will be used in both stages and that the vehicle will be unmanned.

Figures 22-29 depict model network layout and address the fourth investigative question in Chapter 1 by detailing how RMLV design decisions and prelaunch activities were incorporated into the simulation model. Due to model size, the Arena model itself cannot be shown here. However, the figures that follow accurately represent the layout of the actual Arena model. These diagrams are similar to what the model validation Delphi study members received during the model review process (see the “Step 7: Validation” section in Chapter 3). Due to the generic nature of the model, some of the diagrams are bypassed, depending on what RMLV processing decisions the user makes. Each ending path in each diagram terminates with a connector symbol that tells the reader which figure comes next in the sequence, if that path was chosen. Figure 22 provides a key to the modules used in Figures 23-29. All other figures are self-explanatory, except for Figure 23, which represents preintegration activities. Preintegration refers to joining the payload to the second stage before the second stage is attached to the first stage. Preintegration assumes that vehicle design and ground support equipment facilitate attaching the preintegrated second stage and payload to the first stage as a single piece. If vehicle design prohibits this, then payload integration will happen in the traditional fashion, after the second stage is mated to the first stage. Preintegration occurs in parallel with maintenance activities in MILEPOST and has the potential to save time during regeneration because payload integration activities are complete before prelaunch operations even start. This assumes that preintegration time will not exceed the time required for RMLV maintenance.

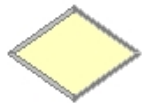
## Module Key



Create Module: The starting point of the entire diagram.



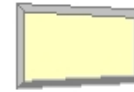
Process Module: An action occurs that takes a certain amount of time.



Decision Module: A decision is made that determines the route to follow.



Separate Module: Indicates the start of parallel processes. The diagram path "splits" at this point, and the processes in each path occur simultaneously.



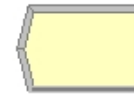
Batch Module: Indicates the end of parallel processes. All processes prior to this module must be complete before the next process can occur.



Submodel Module: Used to provide a clear overview of the entire diagram. Each submodel is a grouping of other modules, for organization purposes.



Connector: Indicates where the path connects on a separate page.

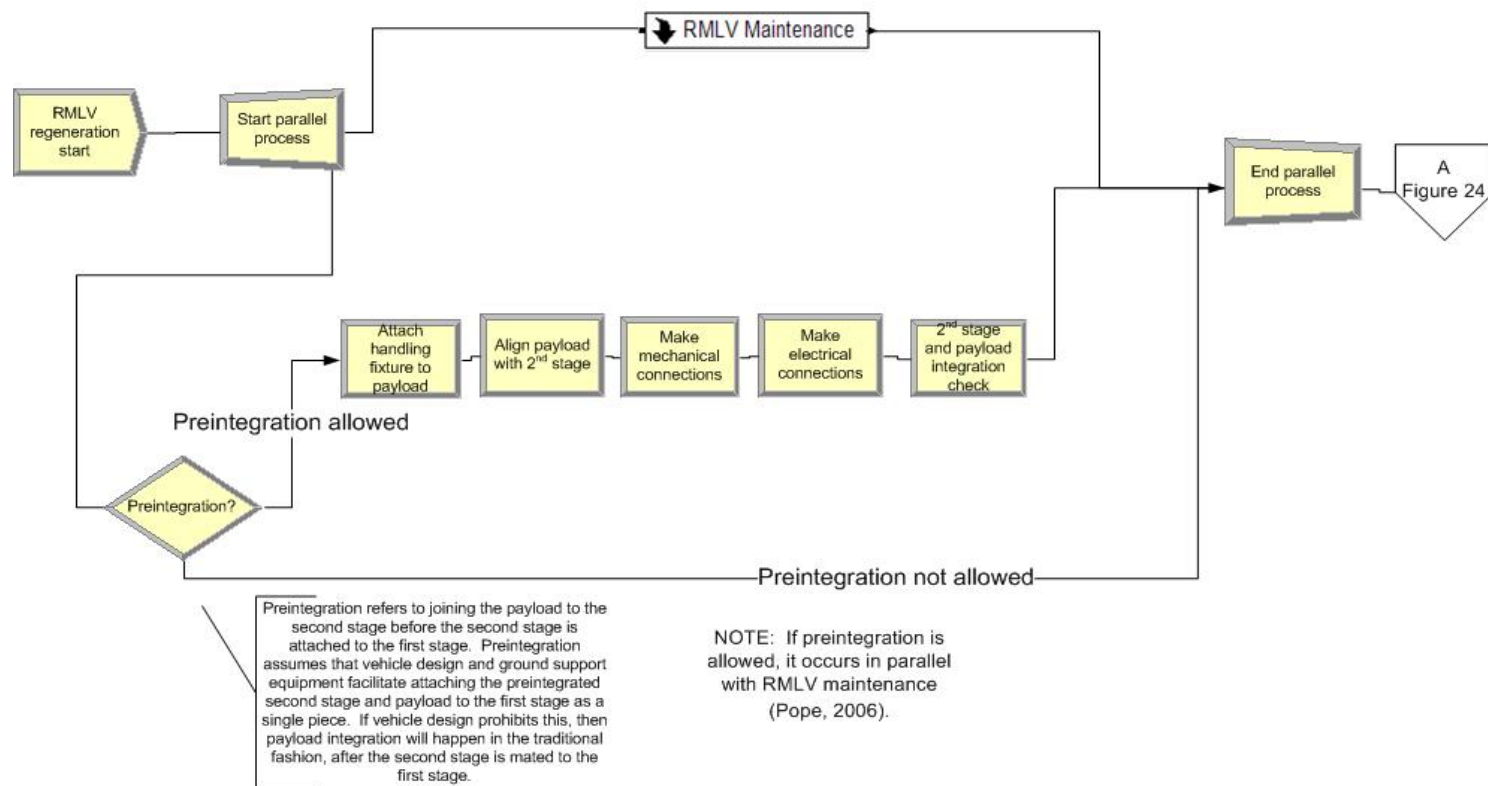


Dispose Module: The ending point of the entire diagram.

Figure 22: Module Key

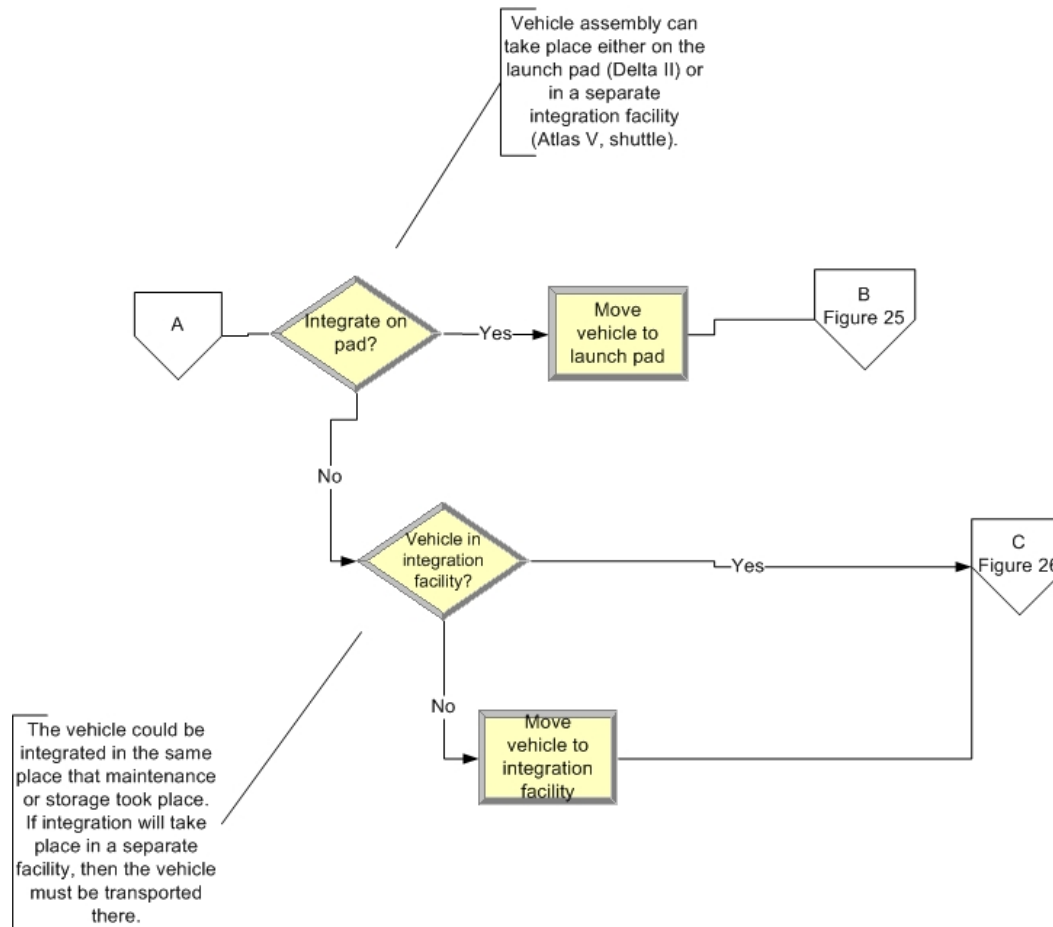


# Preintegration



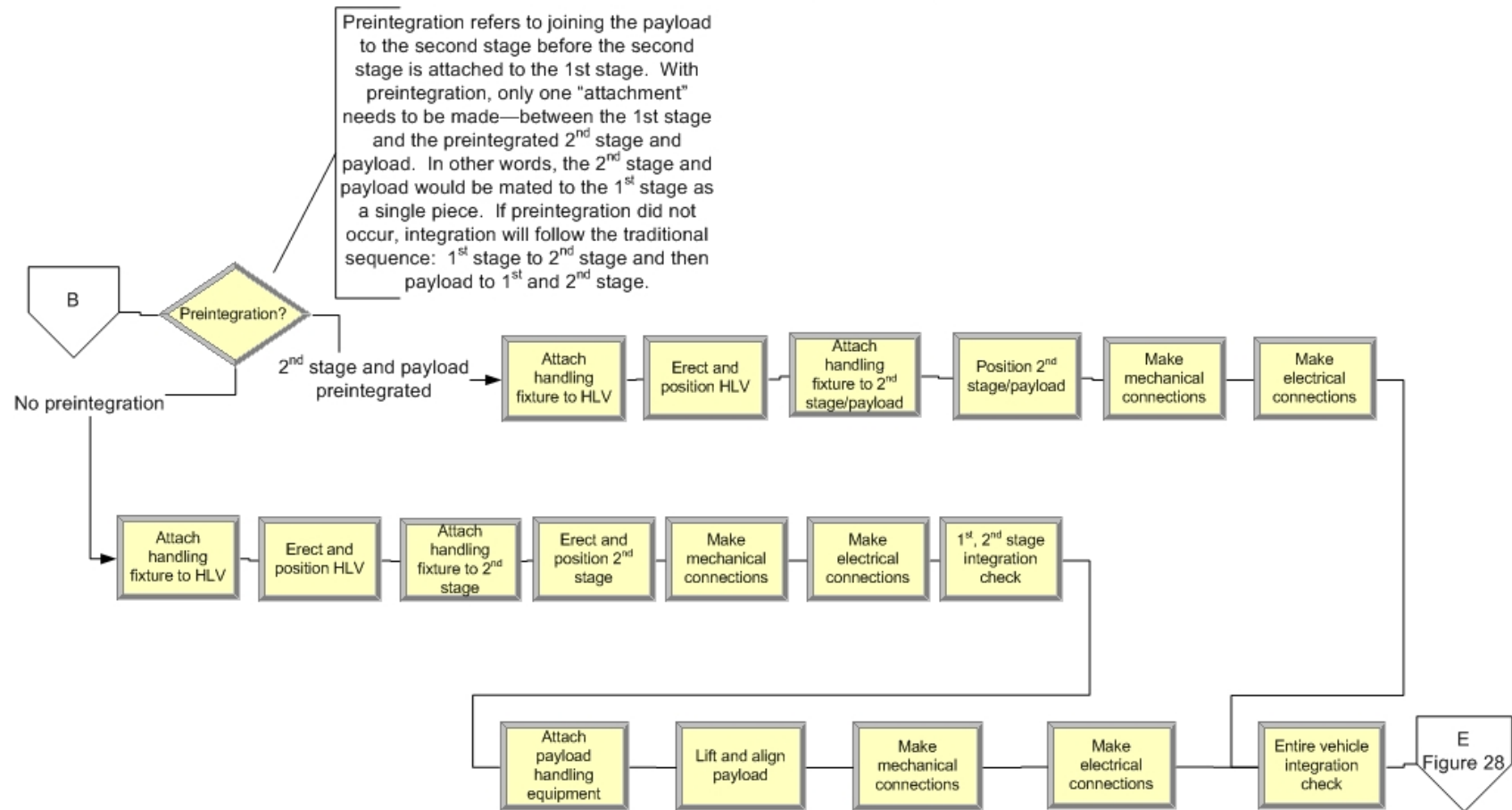
**Figure 23: Preintegration**

## Vehicle Integration Preliminary Considerations

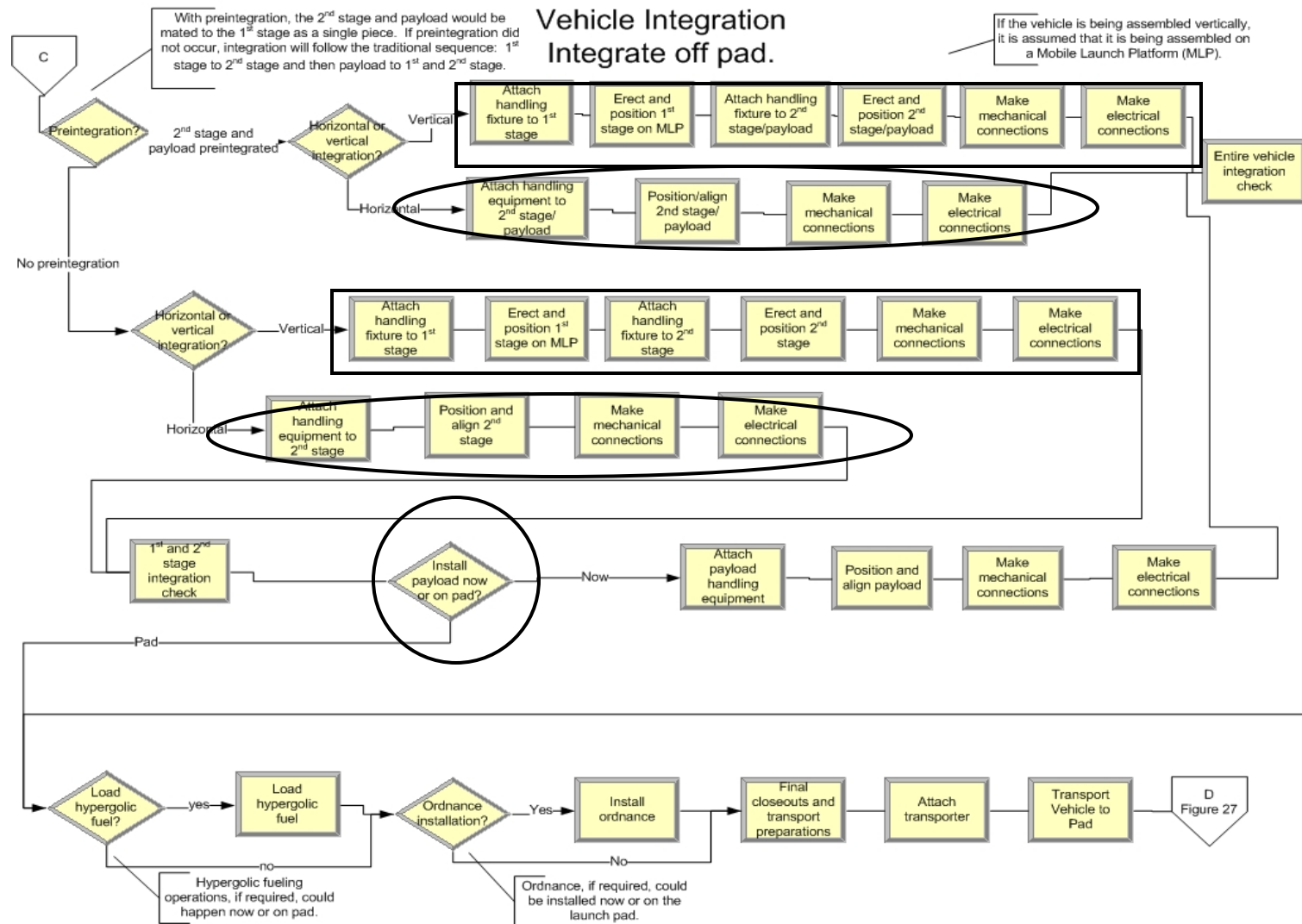


**Figure 24: Vehicle integration preliminary decisions**

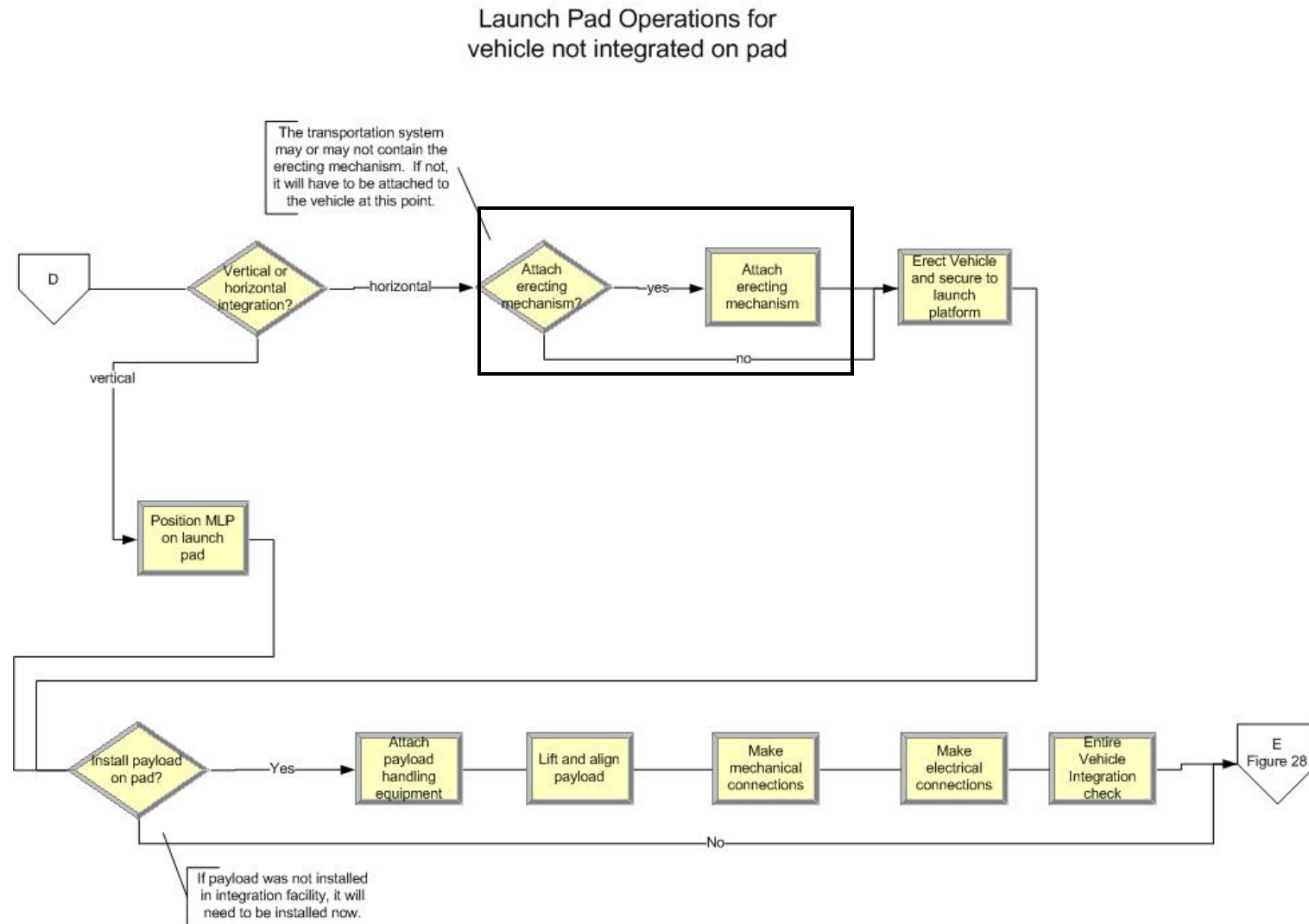
## Vehicle Integration Integrate on pad.



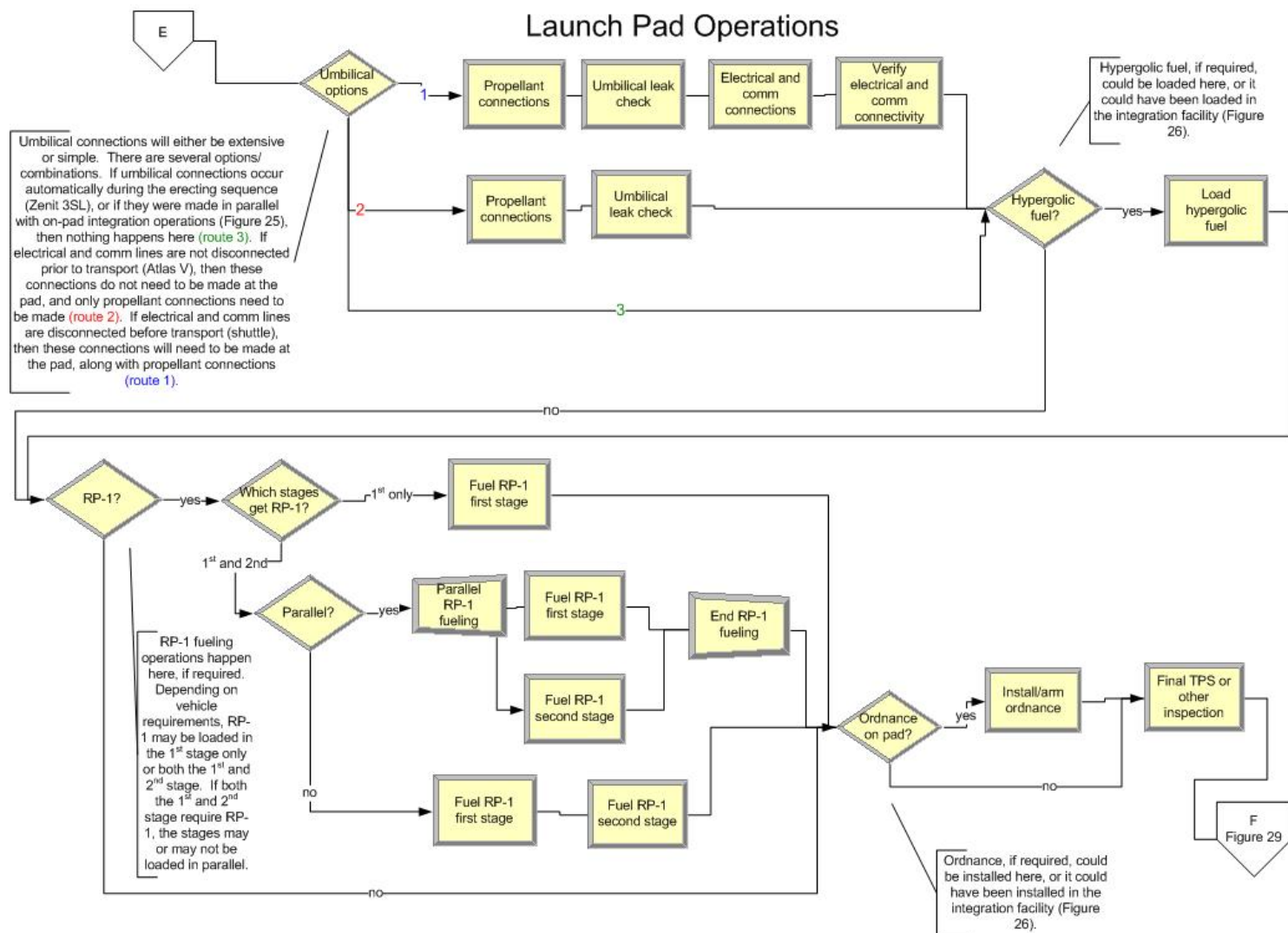
**Figure 25: On-pad integration operations**



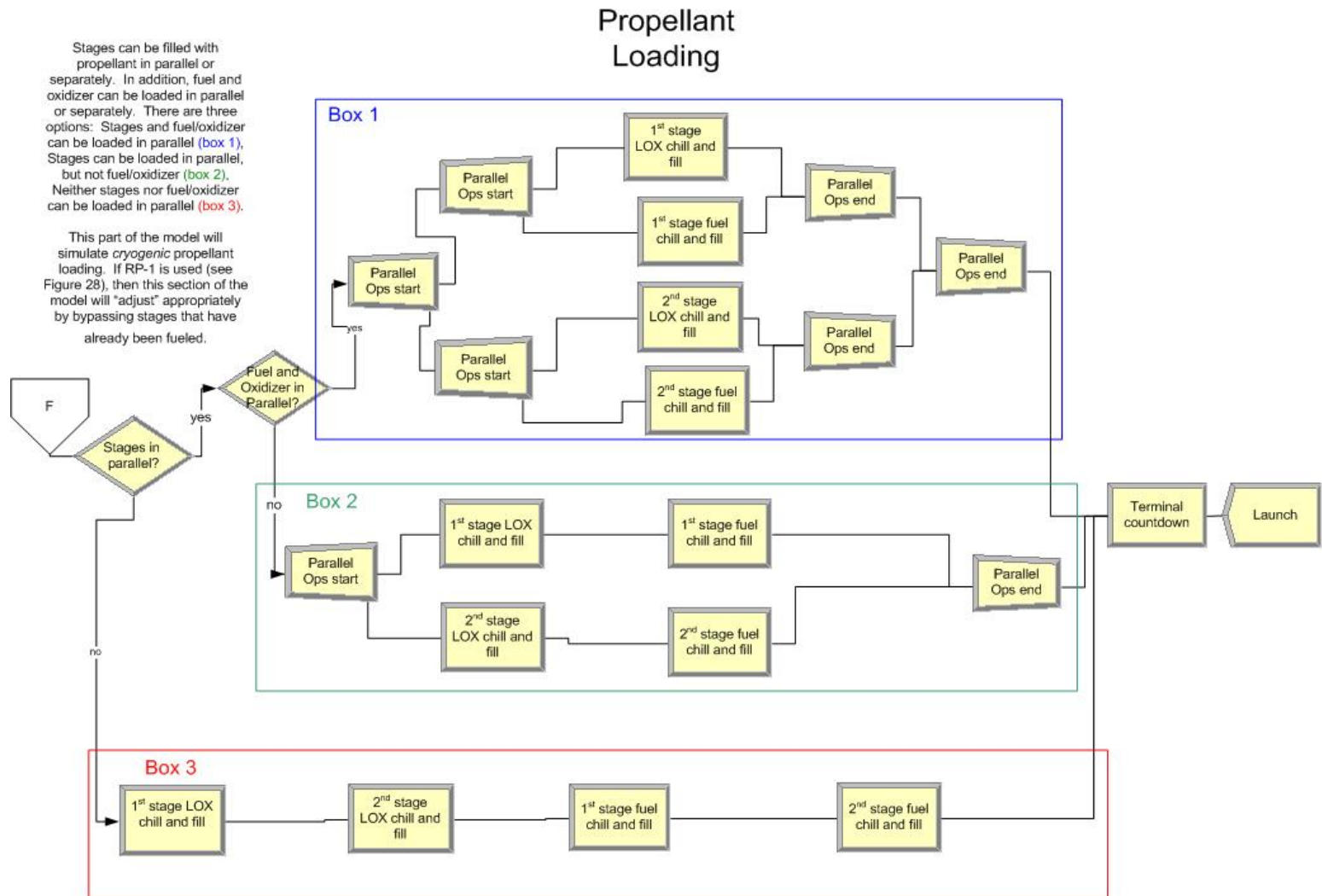
**Figure 26: Off-pad integration operations**



**Figure 27: Initial launch pad operations for vehicle integrated off pad**



**Figure 28: Launch pad operations**



**Figure 29: Propellant loading operations**

The model represented by Figures 22-29 was developed by comparing prelaunch operations for existing launch vehicles. Atlas V, Delta IV, Zenit, and shuttle were the primary systems compared for this purpose. The author noted the many different processing options used by these systems and incorporated these options into his model. The next seven paragraphs explain the link between the model's key processes and the existing systems the processes were based upon.

Figure 23 describes preintegration operations. Although different from the type of preintegration described in this figure, the shuttle utilizes a preintegration concept as well. The shuttle's solid rocket boosters are assembled and attached to the external tank before the integration of the orbiter. Once orbiter maintenance is complete, the orbiter is integrated to the preintegrated solid rocket boosters and external tank (Cates, 2003:82-112). This allows much of the time-consuming integration operations to be completed before shuttle prelaunch operations (as defined by this thesis on page 27) begin.

Figure 24 depicts the integration location decision. This point in the model responds to the user's integration location choice and directs the entity to either on-pad or off-pad integration operations. On-pad integration has been used by many ELVs in the past, but both Evolved Expendable Launch Vehicles have moved toward more off-pad integration operations. Atlas V undergoes stage and payload integration off the launch pad in a vertical integration facility.<sup>3</sup> Delta IV undergoes stage integration off the launch pad in a horizontal integration facility but does not integrate its payload until after it is erected at the launch pad.

---

<sup>3</sup> This is true for Atlas V operations at Cape Canaveral Air Force Station, but Atlas V integration at Vandenberg AFB takes place on the launch pad (*Atlas Launch System Mission Planner's Guide*, 2004:7-10).



Figure 26 includes both horizontal and vertical integration operations. Vertical integration activities (the boxed processes in Figure 26) are based upon Atlas V integration operations. Horizontal integration activities (designated by the ovals in Figure 26) are based upon Delta IV and Zenit integration operations.

The circled decision module in Figure 26 refers to the payload integration location decision. Even if stage integration takes place off the launch pad, payload integration could still take place on the launch pad, as demonstrated by Delta IV. Off-pad payload integration is based upon the Atlas V, and on-pad payload integration (see Figure 27) is based upon the Delta IV.

The boxed modules in Figure 27 refer to an erecting mechanism option inspired by the Zenit 2. The Zenit 2 is transported to the launch pad on a transporter that also includes the erecting mechanism (Isakowitz and others, 2004:557). If the erecting mechanism is not built into the vehicle transporter, then it will have to be attached to the vehicle once it gets to the launch pad.

Figure 28 includes three umbilical attachment options. The first option is based upon a shuttle-like umbilical attachment process that necessitates propellant and electrical and communication connections at the launch pad. The second option is based upon the Atlas V, which only requires a few simple propellant connections at the launch pad. Atlas V retains electrical and communication connectivity during transport to the launch pad, so these attachments do not have to be reconnected once the vehicle arrives at the launch pad (Centore, 2005). Option three is based upon the Zenit 3SL, which completes all umbilical attachments automatically during the erection sequence (Isakowitz and others, 2004:558).

The boxed activities in Figure 28 represent RP-1 loading operations in either the first stage only or the first stage and the second stage. These activities are based upon Atlas V and Zenit operations. Atlas V uses RP-1 only in the first stage (Centore, 2005). Zenit uses RP-1 in its first and second stage (Isakowitz and others, 2004:550).

## **Model Verification Results**

The Assertion Checking method used to verify this model employed record modules strategically placed throughout the model to keep track of entity behavior. Expected record module statistics were compared with actual record module statistics to verify that entities followed the intended path through the model. Entity flow is directed by decision modules, which in turn are controlled by initial variable values designated via user-defined choices in the graphical user interface. The combination of decisions the user makes determines the path the entity will take through the model. The main purpose of the Assertion Checking verification scheme was to ensure that entities were following the exact path designated by the user. The decisions to be made by the user are given in Table 2. The “Decision” column lists all the different decisions a user will make before running the model. These decisions address the third investigative question in Chapter 1 by listing the different RMLV “design drivers” and processing options that will have an effect upon the number, type, and duration of RMLV prelaunch activities. The next three columns display the options associated with each decision. Most decisions have only two options available, but some have three. The “Result” column uses the Excel RANDBETWEEN function in each row to randomly make the decision. For example,

**Table 2: RMLV design and processing decisions**

	<b>Decision</b>	<b>Option 1</b>	<b>Option 2</b>	<b>Option 3</b>	<b>Result</b>
1	Preintegration	Preintegration allowed	Preintegration not allowed		2
2	Integration location	Integrate on launch pad	Integrate off launch pad		1
3	Off pad integration location	Off-pad integration in maintenance bay	Off-pad integration in separate integration facility		N/A
4	Integration orientation	Stages integrated vertically	Stages integrated horizontally		N/A
5	Location of payload integration	Integrate payload in integration facility	Integrate payload on launch pad		N/A
6	Hypergolic fuels	Hypergolic fuels not required	Hypergolic fuels required		1
7	Hypergolic loading location	Load hypergolic fuels in integration facility	Load hypergolic fuels on launch pad		N/A
8	Ordnance	Ordnance not required	Ordnance required		2
9	Ordnance installation location	Install ordnance in integration facility	Install ordnance on launch pad		2
10	Erecting mechanism	Mechanism attached at launch pad	Mechanism part of vehicle transporter		N/A
11	Umbilicals	Propellant and electrical connections required	Propellant connections required	No separate umbilical connections required	3
12	RP-1	Vehicle uses RP-1	Vehicle does not use RP-1		1
13	RP-1 in which stages	Only the first stage uses RP-1	The first and second stage use RP-1		1
14	Parallel RP-1 operations	Stages can be loaded with RP-1 in parallel	Stages cannot be loaded with RP-1 in parallel		N/A
15	Parallel cryogenic operations	Fuel and oxidizer and stages can be loaded in parallel	Stages can be loaded in parallel, but not fuel and oxidizer	Neither stages nor fuel and oxidizer can be loaded in parallel	2

for a decision with two options available, the associated cell in the “Results” column would use the following formula:

$$= \text{RANDBETWEEN}(1,2) \quad (2)$$

If the formula produces a 1, option 1 is chosen. If the formula produces a 2, option 2 is chosen. By employing this method in each row, Excel randomly generates an entity path in the “Results” column. Notice that some cells in the “Results” column contain “N/A.” This is because some decisions do not need to be considered depending upon initial decisions made earlier in the model. For instance, if the integration location decision returns option 1 (on-pad integration) then the next three rows will return “N/A” because they refer to off-pad integration decisions that no longer have to be made.

The author randomly generated and tested 50 separate entity paths. Tables 3 and 4 display the results of the first verification test. Table 3 denotes the specific entity path that was tested.

**Table 3: Entity path for first verification test**

Decision	Path
Preintegration	1
Integration location	2
Off pad integration location	2
Integration orientation	1
Location of payload integration	N/A
Hypergolic fuels	2
Hypergolic loading location	1
Ordnance	1
Ordnance installation location	N/A
Erecting mechanism	N/A
Umbilicals	2
RP-1	2
RP-1 in which stages	N/A
Parallel RP-1 operations	N/A
Parallel cryogenic operations	1

**Table 4: Record module expected versus actual values**

<b>Record Modules</b>	<b>Expected</b>	<b>Actual</b>
Preintegration	1	1
No Preintegration	0	0
Preintegration 1	0	0
No Preintegration 1	0	0
Preintegration 2	1	1
No Preintegration 2	0	0
On Pad Int	0	0
Off Pad Int	1	1
Mx Bay	0	0
Integration Facility	1	1
Vertical	1	1
Horizontal	0	0
Vertical 1	0	0
Horizontal 1	0	0
Horizontal 2	0	0
Vertical 2	1	1
PId In Int Facility	0	0
PId On Pad	0	0
Hypers off pad	1	1
No Hypers	0	0
Hypers on pad	0	0
No Hypers 1	1	1
Ordnance off pad	0	0
No ordnance	1	1
Ordnance on pad	0	0
No ordnance 1	1	1
Attach Erect Mech	0	0
Erecting Mech Built In	0	0
No Umbilical	0	0
Prop Umbilical	1	1
Prop and Elec Umbilical	0	0
RP None	1	1
RP used	0	0
RP 1st and 2nd	0	0
RP 1st	0	0
RP Parallel	0	0
RP Serial	0	0
Parallel stage and prop.	1	1
Parallel stage	0	0
Serial cryo	0	0

Table 4 gives a list of all the record modules within the model in the “Record Modules” column. Each record module simply counted the number of entities that passed through it. Since the model only sends one entity through at a time, each record module returns a value of either 1 or 0, depending on whether or not the entity passed through that module. The “Expected” column lists the expected record module statistics that correlate with the entity path tested, and the “Actual” column lists the actual record module statistics after one model replication. In this case, the “Expected” and “Actual” columns match, which means entity flow progressed through the model according to the user’s decisions. Each of the remaining 50 entity paths that were tested resulted in matching “Expected” and “Actual” columns.

The author determined the number of verification tests to complete by using a Binomial Acceptance Testing Plan (Ebeling, 2005:316, 317). Although this approach is usually used to test the reliability of physical systems such as machines, vehicles, or components, it has applicability to the model verification tests performed for this thesis. The purpose of the test is to demonstrate a certain system reliability with an acceptable level of confidence. In this case, the “system” under consideration is the simulation model. Reliability as it refers to physical systems is usually defined as “the probability of nonfailure over time” (Ebeling, 2005: 5). As it applies to this simulation model, it could be defined as “the probability the model will not fail for a randomly generated entity path.” The binomial acceptance plan completes a total of  $n$  tests and observes  $X$  test failures amongst those  $n$  tests. If  $R$  is the true model reliability for each test,  $X$  has a binomial probability distribution with parameters  $n$  and  $p = (1 - R)$ . The binomial test plan specifies the sample size  $n$  and the maximum number of failures,  $r$ , allowed to

confidently state that  $R = R_1$ , the desired system reliability. If  $X \leq r$ , then it is concluded that  $R = R_1$ . If  $X > r$ , then it is concluded that  $R = R_2 < R_1$ .

Due to the randomness of sampling, it is possible to come to a false conclusion concerning the true value of  $R$ . To lower the probability of this, it is necessary to minimize both Type I ( $\alpha$ ) and Type II ( $\beta$ ) error. Type I error is the probability of incorrectly rejecting the null hypothesis that  $R = R_1$ . Type II error is the probability of incorrectly failing to reject the null hypothesis that  $R = R_1$ . The relationship between  $\alpha$  and  $\beta$  and  $n$  and  $r$  is expressed in the following two adaptations of the binomial probability mass function:

$$\sum_{i=0}^r \binom{n}{i} (1-R_1)^i R_1^{n-i} = 1 - \alpha \quad (3)$$

$$\sum_{i=0}^r \binom{n}{i} (1-R_2)^i R_2^{n-i} = \beta \quad (4)$$

The challenge is to find values for  $n$  and  $r$  that result in acceptable values of  $\alpha$  and  $\beta$ . The acceptance plan for this model used one of the selected reliability acceptance plans given by Ebeling (Ebeling, 2005:317). The values associated with this plan are given in Table 5.

**Table 5: Selected reliability acceptance plan (Ebeling, 2005:317)**

$R_1$	$R_2$	$n$	$r$	$1 - \alpha$	$\beta$
.99	.90	50	1	.911	.034

This particular acceptance plan states that it can be concluded with 91.1 percent confidence that  $R = .99$  if there is only 0 or 1 failure out of 50 trials. For the author's prelaunch operations model, 50 tests resulted in 0 failures. As a result, the author can be

91.1 percent confident that  $R$  for the model = .99. As a result of this verification effort, users can have confidence in a credible model that accurately represents the modeler's intentions.

## **Experimental Design Results**

The author used model output to complete three experiments pertaining to RMLV regeneration time. The purpose of these experiments is not to provide an exact estimate of RMLV regeneration time but rather to provide preliminary insights concerning several RMLV prelaunch operations processing options. As such, the experiments answer the fifth investigative question in Chapter 1. The experiments were carried out using output data from MILEPOST so that the experiment results could be stated in terms of total RMLV regeneration time instead of just prelaunch operations time. However, the experiments only analyzed prelaunch operations alternatives and did not explore maintenance sensitivities, because these are addressed in concurrent research by Pope (Pope, 2006). The following questions reflect the three experiments:

1. How does the decision concerning whether or not to allow preintegration of the 2<sup>nd</sup> stage and payload affect RMLV regeneration time?
2. How does integration location affect RMLV regeneration time?
3. How does integration orientation (vertical versus horizontal integration) affect RMLV regeneration time?

Each of these three questions was answered by comparing MILEPOST output data for two different processing scenarios. Output comparisons were made using a large sample confidence interval for a difference in means. By forming a confidence interval



around the difference between two average RMLV regeneration times, the author was able to conclude whether the ground processing alternative in question has an effect upon RMLV regeneration time. The formula for a large sample confidence interval for a difference in means is given by

$$(\bar{x}_1 - \bar{x}_2) \pm z_{\alpha/2} \sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}} \quad (5)$$

Several conditions are required for using this confidence interval to make inferences. First, the two samples must be independently and randomly selected from two target populations. Arena's random number generation scheme satisfies the random selection requirement. In comparing two RMLV processing scenarios, Arena completes two simulation runs, and the output from one simulation run is not dependent upon output from the other simulation run. This satisfies the independence requirement. Second, both samples must be sufficiently large ( $n_1 \geq 30$  and  $n_2 \geq 30$ ) to guarantee that the sampling distribution of the difference in means approximates the normal distribution (McClave and others, 2005:483). 50 replications were simulated for each processing scenario to meet this requirement.

The "Data Collection" section in Chapter 3 discusses how distributions were formed to populate the model's process modules. Appendix C contains the complete list of distributions that were used in each of these experiments. While these distributions represent plausible approximations for what RMLV activity durations may be in the future, they are still only estimates and may or may not represent future RMLV operations. The results of the experiments below must be analyzed with this caveat in mind.

### ***Experiment 1: Effect of Preintegration Decision***

The null and alternative hypotheses associated with this experiment are given below.

$H_0$ : RMLV average regeneration time without preintegration – RMLV average regeneration time with preintegration = 0.

$H_a$ : RMLV average regeneration time without preintegration – RMLV average regeneration time with preintegration > 0.

To test this hypothesis, two entity paths were simulated. Table 6 gives the two entity paths that were compared. The “Preintegration” column gives the entity path that represents RMLV prelaunch operations with preintegration.

**Table 6: Entity paths compared for the preintegration experiment**

<b>Decision</b>	<b>Preintegration</b>	<b>No Preintegration</b>
Preintegration	1	2
Integration location	2	2
Off pad integration location	2	2
Integration orientation	2	2
Location of payload integration	N/A	1
Hypergolic fuels	1	1
Hypergolic loading location	N/A	N/A
Ordnance	2	2
Ordnance installation location	2	2
Erecting mechanism	1	1
Umbilicals	1	1
RP-1	1	1
RP-1 in which stages	1	1
Parallel RP-1 operations	N/A	N/A
Parallel cryogenic operations	2	2

The “No Preintegration” column gives the entity path that represents RMLV prelaunch operations without preintegration. Notice that besides the preintegration decision, every other decision was held constant, so that the preintegration decision could be analyzed by itself apart from interaction from the other decisions. Table 2 shows what options are represented by the values in the “Preintegration” and “No Preintegration” columns.

Output statistics for the two scenarios are given in Table 7.

**Table 7: Preintegration experiment output statistics**

<b>Scenario</b>	<b>Mean Regeneration Time (hours)</b>	<b>Standard Deviation (hours)</b>
Preintegration	68.50	4.72
No Preintegration	72.33	6.59

Using Equation 5 and  $\alpha = .05$ , the values in Table 7 result in the following confidence interval for the difference in RMLV mean regeneration time:

$$[1.59, 6.08] \text{ hours}$$

Since this confidence interval does not contain zero, the null hypothesis is rejected. It can be concluded with 95% confidence that preintegration does decrease RMLV mean regeneration time for RMLV ground operations as represented by MILEPOST.

### ***Experiment 2: Effect of Integration Location Decision***

The null and alternative hypotheses associated with this experiment are given below.

$H_0$ : RMLV average regeneration time for integration off the launch pad – RMLV average regeneration time for integration on the launch pad = 0.

$H_a$ : RMLV average regeneration time for integration off the launch pad – RMLV average regeneration time for integration on the launch pad  $> 0$ .

Table 8 gives the two entity paths compared for this experiment. The “Integrate on Launch Pad” column in Table 8 gives the entity path that represents vehicle integration on the launch pad. The “Integrate off Launch Pad” column gives the entity path that represents vehicle integration off the launch pad. For this experiment, on-pad integration operations, where integration by necessity occurs vertically, were compared to off-pad integration operations that were also performed in the vertical orientation. Output statistics for the two scenarios are given in Table 9.

**Table 8: Entity paths compared for the integration location experiment**

<b>Decision</b>	<b>Integrate on Launch Pad</b>	<b>Integrate off launch pad</b>
Preintegration	1	1
Integration location	1	2
Off pad integration location	N/A	2
Integration orientation	N/A	1
Location of payload integration	N/A	1
Hypergolic fuels	1	1
Hypergolic loading location	N/A	N/A
Ordnance	2	2
Ordnance installation location	2	2
Erecting mechanism	N/A	1
Umbilicals	1	1
RP-1	1	1
RP-1 in which stages	1	1
Parallel RP-1 operations	N/A	N/A
Parallel cryogenic operations	2	2

**Table 9: Integration location experiment output statistics**

<b>Scenario</b>	<b>Mean Regeneration Time (hours)</b>	<b>Standard Deviation (hours)</b>
Integrate on launch pad	68.99	4.90
Integrate off launch pad	72.77	4.61

Using Equation 5 and  $\alpha = .05$ , the values in Table 9 result in the following confidence interval for the difference in RMLV mean regeneration time:

$$[1.91, 5.64] \text{ hours}$$

Since this confidence interval does not contain zero, the null hypothesis is rejected. It can be concluded with 95% confidence that integration on the launch pad does decrease RMLV mean regeneration time for RMLV ground operations as represented by MILEPOST. This difference is primarily due to the exclusion of certain activities that are required by off-pad integration but not by on-pad integration. For instance, if integration takes place on the launch pad, time does not have to be spent on transportation activities from the integration facility to the launch pad. In addition, with on-pad integration, time does not need to be spent erecting the vehicle or securing the mobile launch platform to the launch pad once the vehicle arrives from the integration facility. While saving time in this way may seem attractive, it comes with a trade-off. Integrating on the launch pad means that the launch pad resource is seized for a longer amount of time. This may not be desirable if there are limited launch pad resources available.

### ***Experiment 3: Effect of Integration Orientation Decision***

The null and alternative hypotheses associated with this experiment are given below.

$H_0$ : RMLV average regeneration time for horizontal integration – RMLV average regeneration time for vertical integration = 0.

$H_a$ : RMLV average regeneration time for horizontal integration – RMLV average regeneration time for vertical integration < 0.

Table 10 gives the two entity paths compared for this experiment. The “Vertical Integration” column in Table 10 gives the entity path that represents vertical integration off the launch pad. The “Horizontal Integration” column gives the entity path that represents horizontal integration off the launch pad. Output statistics for the two scenarios are given in Table 11.

**Table 10: Entity paths compared for the integration orientation experiment**

Decision	Vertical Integration	Horizontal Integration
Preintegration	2	2
Integration location	2	2
Off pad integration location	2	2
Integration orientation	1	2
Location of payload integration	1	1
Hypergolic fuels	1	1
Hypergolic loading location	N/A	N/A
Ordnance	2	2
Ordnance installation location	2	2
Erecting mechanism	N/A	1
Umbilicals	1	1
RP-1	1	1
RP-1 in which stages	1	1
Parallel RP-1 operations	N/A	N/A
Parallel cryogenic operations	2	2

**Table 11: Integration orientation experiment output statistics**

<b>Scenario</b>	<b>Mean Regeneration Time (hours)</b>	<b>Standard Deviation (hours)</b>
Vertical Integration	75.55	6.61
Horizontal Integration	72.33	6.59

Using Equation 5 and  $\alpha = .05$ , the values in Table 11 result in the following confidence interval for the difference in RMLV mean regeneration time:

$$[-6.56, -.63] \text{ hours}$$

Since this confidence interval does not contain zero, the null hypothesis is rejected. It can be concluded with 95% confidence that horizontal integration instead of vertical integration decreases RMLV mean regeneration time for RMLV ground operations as represented by MILEPOST.

### **Summary**

This chapter began with a description of the prelaunch operations model that was developed for this thesis. The chapter then covered model verification results and concluded with a description of the experimental design and associated statistical analysis. The following chapter offers research conclusions, research limitations, and suggestions for further study.

## **V. Conclusions and Recommendations**

### **Introduction**

This chapter begins with a research summary and then offers research conclusions. Next, research limitations are discussed. The chapter concludes with suggestions for follow-on research.

### **Research Summary**

The prelaunch operations model developed for this thesis is a useful tool that can help the Air Force understand how different RMLV design and processing decisions affect RMLV regeneration time. The validation and verification methods employed for this thesis ensure that the model is a credible representation of future RMLV prelaunch operations, based upon the knowledge and data that is currently available.

Model validation was accomplished via a Delphi study intended to elicit expert opinion on the model conceptual flow network. 15 panel members were chosen from a variety of organizations, including the Air Force Research Laboratory Air Vehicles Directorate, Aeronautical Systems Center, NASA, and Air Force Space Command. Three Delphi “rounds” were completed. In each round, a model conceptual flow diagram was sent to the panel members, and panel member responses were collected. The author adjusted the model network based upon the advice he received from the panel members. The Delphi study produced many good suggestions that helped the author create a model that accurately represents RMLV prelaunch operations, to the extent possible due to limited knowledge about future RMLV design.



Model verification was accomplished via an Assertion Checking (Defense Modeling and Simulation Office, 1996: 4-13) method that compared model developer intent to actual model operation. 50 entity paths were randomly generated and tested in the model. Using record modules placed throughout the model, actual entity flow was captured and compared to intended entity flow. For all 50 tests, actual entity flow matched intended entity flow.

This prelaunch operations simulation model can be considered a “baseline” model that can be built upon and modified in the future. As such, it is a good starting point for RMLV processing analysis, but the model will only improve as more information about future RMLV operations becomes available.

## **Research Conclusions**

After validation and verification were complete, the model was used to evaluate several different prelaunch operations processing options. Three experiments were conducted. For the first experiment, RMLV operations with preintegration of the second stage and payload were compared to RMLV operations without preintegration of the second stage and payload. The result of the experiment supported the hypothesis that preintegration reduces RMLV regeneration time. The second experiment compared integration on the launch pad to integration off the launch pad. The outcome of the experiment suggested that on-pad integration, as opposed to off-pad integration, could reduce RMLV regeneration time. It must be noted that this experiment is subject to the limitations of the model, which only allows one “mission entity” to enter the model per replication. As such, the model in its current form does not simulate steady-state RMLV

operations. Under steady state RMLV operations, on-pad integration may be undesirable because it ties up the launch pad resource for longer periods of time. The final experiment compared vertical integration operations to horizontal integration operations. The outcome of the experiment suggested that horizontal integration could result in shorter regeneration times than vertical integration.

In developing the conceptual model of RMLV prelaunch operations, the author gained a great deal of knowledge and insight into different RMLV prelaunch processing options, how they interact, and how they may affect regeneration time. Using this knowledge and insight, the author developed a list of suggestions concerning RMLV prelaunch operations:

1. *Utilize a preintegration approach if possible.*

Integrating the second stage with the payload while first stage maintenance is taking place results in fewer integration activities that have to occur once maintenance is complete. Instead of integrating the first stage with the second stage and then mating the payload to the second stage, the preintegrated second stage and payload could be attached to the first stage as a single piece.

2. *Integrate off the launch pad if few launch pads are available.*

If the number of usable launch pads is high, then RMLV regeneration time could be decreased by integrating on the launch pad. However, if the number of launch pads is limited, then it may be best to integrate off the launch pad to keep launch pads free for launch operations. If a launch pad is used for integration and launch operations, the pad will be occupied for a longer amount of time than if it is used only for launch operations.

The more time a launch pad is occupied by a single vehicle, the less likely it will be that the launch pad will be available when another vehicle needs it.

*3. Integrate horizontally if possible.*

With horizontal integration, vehicle access is easier, which, in theory, should result in shorter integration times. In addition, horizontal integration is desirable because personnel and vehicle safety is degraded with vertical integration. Extra safety precautions need to be taken with vertical integration due to the dangers associated with falling tools and falling personnel.

*4. Avoid the use of hypergolic fuels and ordnance.*

Both hypergolic fueling and ordnance installation operations increase regeneration time significantly. In addition, their use presents additional safety hazards for personnel.

*5. Build the erecting mechanism into the vehicle transporter.*

A horizontally integrated vehicle will have to be erected once it gets to the launch pad. If the erecting mechanism is built into the transport platform, time will not have to be taken to attach the erecting mechanism once the vehicle reaches the launch pad.

*6. Streamline umbilical attachments as much as possible.*

There are a variety of different umbilical alternatives that can decrease umbilical attachment time. Whenever umbilicals are connected, proper umbilical attachment must be verified. This means that disconnecting and reconnecting umbilical attachments should be avoided. In addition, automated umbilical attachments can decrease umbilical connection time. It also stands to reason that the more umbilical attachments that need to

be made and the more complex those attachments are, the more time will be spent in making those attachments.

*7. Utilize parallel propellant loading.*

Loading fuel and oxidizer in both stages simultaneously has the potential to speed propellant loading operations. Fill rate and stage filling sequence will be influenced by vehicle structure limitations, but outside of these requirements, parallel propellant loading should be used to the maximum extent possible.

In addition to the seven recommendations given above, the author made several other observations throughout the course of this research that may be of value to those making RMLV acquisition decisions. First, the activities required for RMLV prelaunch operations present a significant difference from aircraft operations. The differences between RMLVs and aircraft were discussed in the “Prelaunch Operations Comparisons” section in Chapter 2. These differences mean that it may be unrealistic to hope that RMLV ground processing will be as simple as aircraft ground processing. RMLVs may never experience the quick regeneration times that aircraft experience. However, this does not mean that the aircraft maintenance community has nothing to offer the RMLV community. This leads to the second observation, which is that RMLV designers and acquisition managers could learn much from proven and efficient aircraft maintenance practices and philosophies. A launch vehicle that requires maintenance is new territory for the Air Force, and the RMLV acquisition community could suffer if it lacks maintenance experience. If the goal is to see future RMLV sortie rates approach aircraft sortie rates, then aircraft maintenance experience will be a valuable asset to the RMLV acquisition and design team. The final observation is that RMLV design alone will not

dictate the speed at which RMLVs can be regenerated. RMLV prelaunch operations require the use of many different types of ground support equipment, and the ease of use and speed of operation of this equipment will have a profound effect upon RMLV regeneration time. For example, vehicle integration will require the use of stage handling fixtures and hoists or cranes. If this equipment operates slowly or is difficult to use, it will slow down integration operations. The importance of ground support equipment means that it is just as important to put thoughtful consideration into its design as it is to put thoughtful consideration into the design of the vehicle itself.

### **Research Limitations**

The most severe limitation of this research stems from the fact that RMLV operations will not commence until several years into the future. This makes it extremely difficult to determine what activities will be included in RMLV prelaunch operations. The model was built by analyzing existing launch vehicle processing flows and then adjusting them to create the RMLV prelaunch operations model. The model is generic enough to accommodate many different processing scenarios, but the eventual RMLV may require activities that were not anticipated. This means that the RMLV prelaunch operations activities included in the model may or may not represent future RMLV operations. While it is difficult to determine what activities will be included in RMLV prelaunch operations, it is perhaps even more difficult to determine how long those activities will take. The activity durations and associated distributions that were used to run the model for the experiments described in Chapter 4 were estimated from similar activities for existing launch vehicles, aircraft, and ICBMs. Until more is known about

RMLV design, it is impossible to know for sure how long prelaunch operations activities will take. This means that the average regeneration times produced by the model should not be considered to be perfect predictions of actual RMLV regeneration time.

Another limitation of the model is that it only accepts one entity per replication. This means that only one “mission,” or one “need to launch” enters into the system per replication. Since each mission entity enters MILEPOST at the beginning of RMLV maintenance operations, each MILEPOST replication essentially represents a regeneration timeline for a single RMLV. In addition, the model only includes ground processing activities and does not simulate the RMLV mission time that takes place between RMLV launch and landing. These limitations mean that the model cannot yet simulate steady state RMLV operations for a fleet of RMLVs. Several model modifications and additions are required to simulate RMLV steady state operations. First, RMLV mission operations, or “mission time” needs to be incorporated. Second, the parallel processes in the model need to be adjusted to properly separate and batch entities if there are multiple entities entering the model. Finally, resources (personnel, equipment, facilities), which are considered unlimited in the author’s model, need to be constrained so that multiple entities can vie for those resources. With these modifications, the model could accommodate multiple mission entities entering the system per some schedule, or randomly per some distribution. This would give the user a sense of not only regeneration time, but also of the sortie rate supported by the system and the utilization rate of resources.

Some other limitations are linked to the assumptions that were made in building the model. For instance, the model assumes two stages—a reusable first stage and an

expendable second stage. The model would have to be modified before it could support analysis of any vehicle configuration other than this. The model also assumes a vertical take off and that both stages will use liquid propellants. Finally, there are no probabilities associated with failed activities in this model. For instance, it is assumed that if an inspection takes place, no discrepancies are found. In addition, there is no probability associated with a scrubbed launch. In other words, it is assumed that each launch goes off when it was intended.

### **Suggestions for Future Research**

There are many ways that this research can be continued. First, future researchers could expand the model to include the entire spectrum of RMLV operations, both on the ground and in space. As stated in the previous section, the model only includes RMLV operations that take place between landing and launch. Modifying the model to include the time between launch and landing would expand the model's capabilities. This would allow a user to analyze steady state RMLV operations over an extended period of time. A follow-on researcher could also add more "bells and whistles" to the model. For instance, the researcher could add probabilities associated with a failed and passed vehicle integration check. The model in its current form assumes that all integration checks are completed without any problems noted.

Any research has room for improvement, and this thesis is no exception. Follow-on researchers could take a second look at the activities included in the model to see if any activities should be added or removed. This type of research will be especially important as time goes on and more knowledge about RMLV operations becomes available.

One interesting study would use computer simulation to analyze how different combinations of numbers of facilities, launch pads, first and second stages, and other resources would affect regeneration time and sortie rate. This issue could be analyzed with a completely new model that would represent only the upper-level view of RMLV operations. The model in this thesis breaks down processes into detailed activities, but for the follow-on research suggested, these detailed activities could be grouped together into one module to represent the upper-level process. Different assumptions could be made concerning durations of regeneration activities, and the simulation would show how these different assumptions affect resource (launch pads, maintenance hangars, etc...) utilization and sortie rate. In addition, different assumptions concerning resource availability could be analyzed to see what effect these assumptions have upon sortie rate and regeneration time. For instance, RMLV operations with one maintenance hangar, one integration facility, and one launch pad could be compared to RMLV operations with two maintenance hangars, two integration facilities, and two launch pads to see how the extra resources affect regeneration time and sortie rate.

The “Research Conclusions” section in this chapter talks about the importance of RMLV designers and acquisition managers learning from aircraft maintenance practices. This presents another opportunity for further research. The “Prelaunch Operations Comparisons” section in Chapter 2 explained the difference between RMLV prelaunch operations and corresponding aircraft operations. More research along these lines needs to be done to analyze which philosophies, policies, and organizational structures within the aircraft sortie generation environment will benefit future RMLV operations. The



researcher pursuing this topic could look for ways that established and successful aircraft maintenance practices could be applied to RMLV maintenance and prelaunch operations.

Finally, as more confidence is gained in the activity durations used to populate the model, more experiments like the ones in Chapter 4 can be done. Each experiment in Chapter 4 only analyzed one isolated processing decision apart from other decisions. Further experiments could look at two or more processing decisions at the same time to see how those decisions interact with each other.

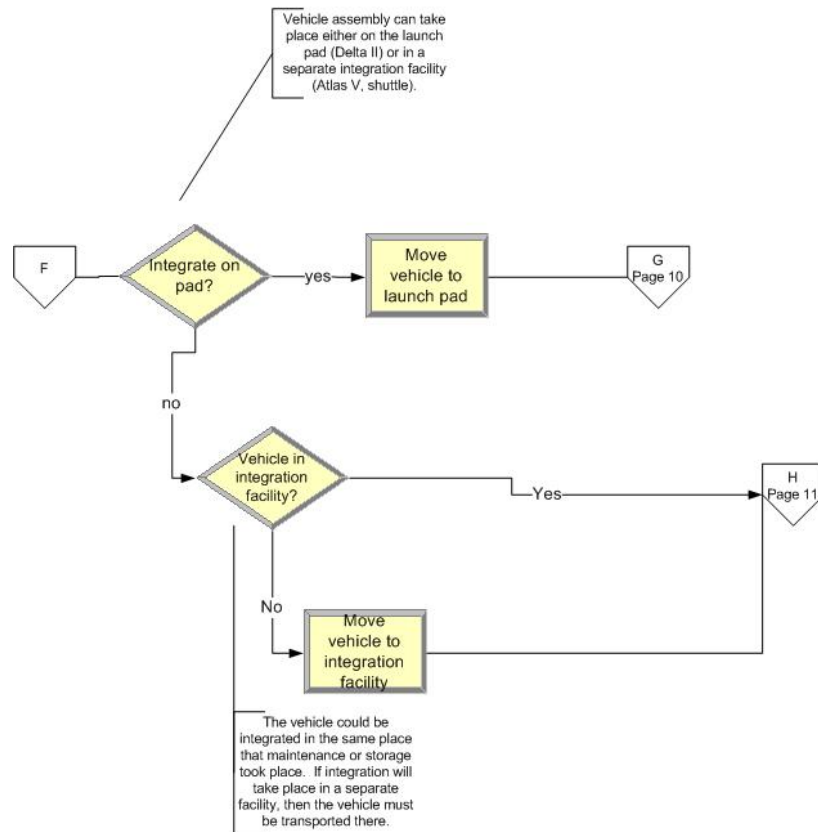
To view the MILEPOST code, see the thesis written by Pope (Pope, 2006).

## **Appendix A: Delphi Study Documentation**

Each of the three Delphi rounds completed for model validation is reproduced here. Each round contains the model diagram that was sent to the Delphi members and their responses for that round. To protect the participant's anonymity, names are not given. Each comment is preceded by an identification number that designates the member who made the comment. The comments are categorized by the model diagram page number the comment refers to. Page numbers are given in the bottom right-hand corner of each page of the model diagram. Notice that the first page in each round is not listed as page number one. This is because the model diagram was sent out together with Pope's maintenance model (Pope, 2006), which appeared before the prelaunch operations model. The Delphi participants looked at RMLV maintenance and prelaunch operations at the same time. Rounds two and three are preceded by short explanations of how the model changed in response to the previous round's comments.

## Round One Model Diagram

### Vehicle Integration Preliminary Considerations

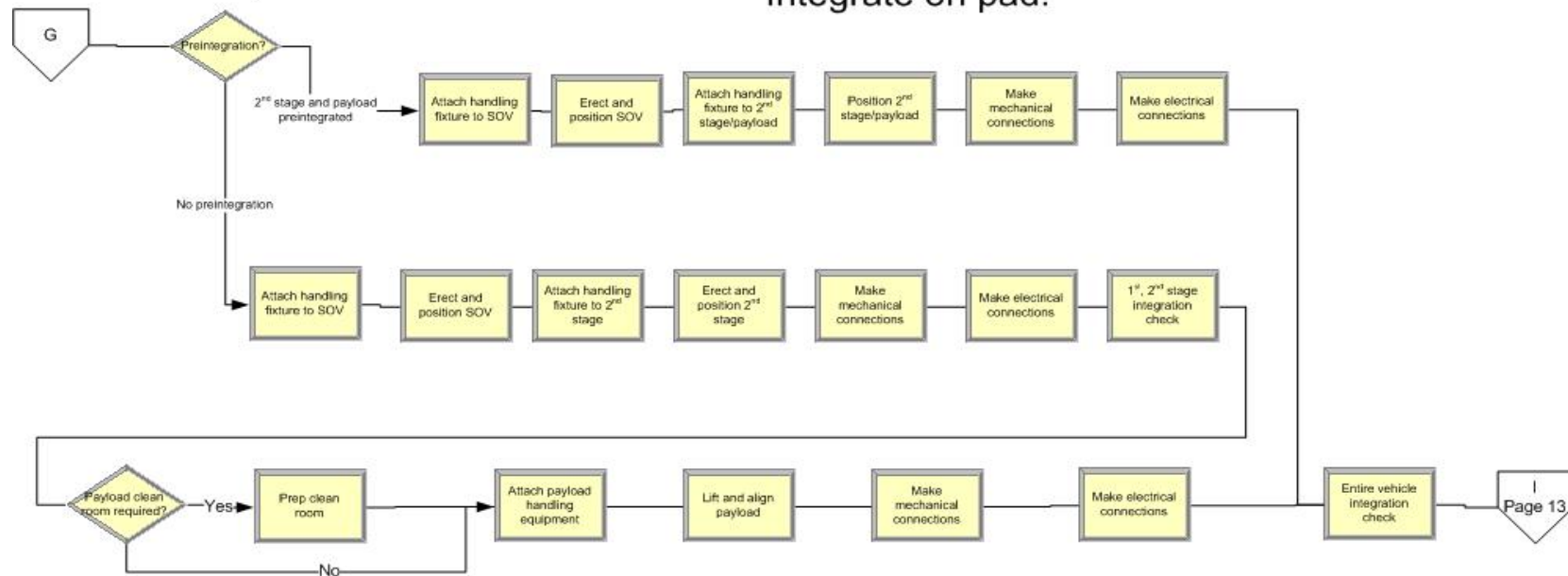


09

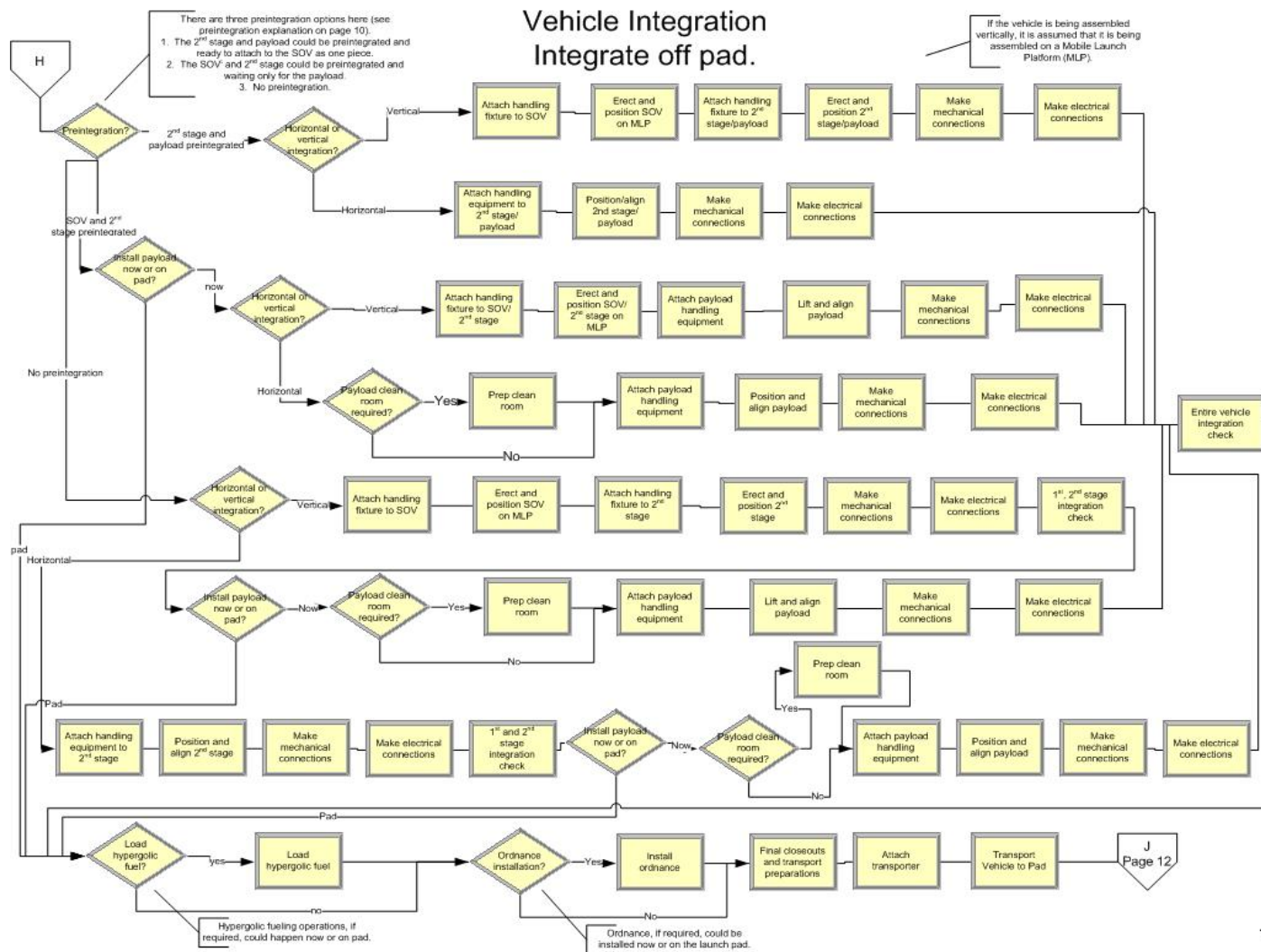
### Round One, Vehicle Integration Preliminary Considerations

Preintegration refers to portions of either vehicle or payload integration taking place before a need to launch arises. Preintegration has the potential to save time in the integration process after a need for launch arises. For this model, it is assumed that preintegration, if any, would have taken place during or prior to the "waiting for need to launch" phase on page 9. There are two preintegration options for integration on pad: First, the 2<sup>nd</sup> stage and payload could be preintegrated and ready to go. This would require only one "attachment" to be made—between the 1<sup>st</sup> stage and the 2<sup>nd</sup> stage/payload. The 2<sup>nd</sup> stage and payload would be mated to the first stage as a single piece. The second option is no preintegration—the second stage must be integrated to the first stage and then the payload must be integrated to the vehicle.

## Vehicle Integration Integrate on pad.

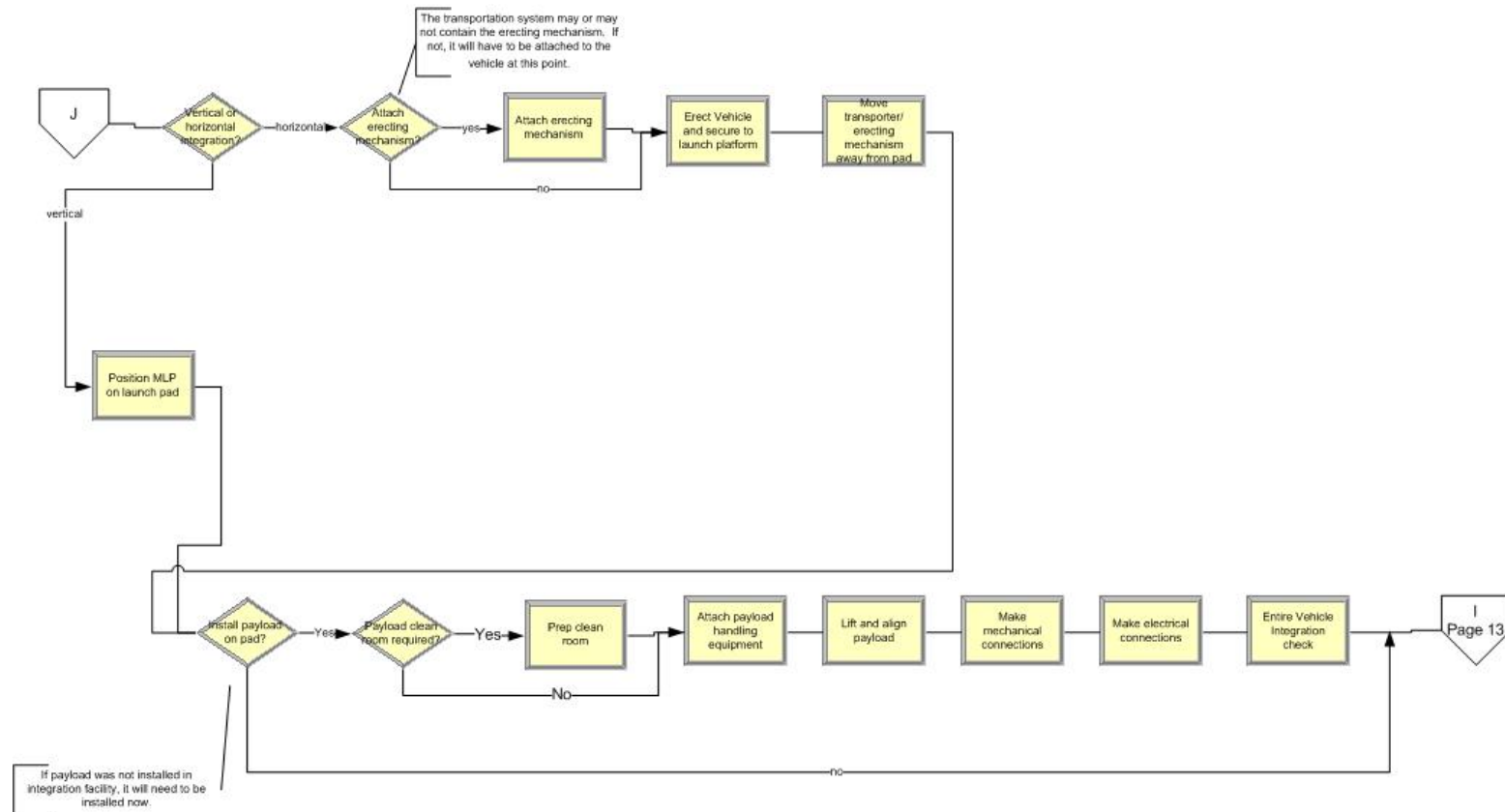


## Round One, On-pad Vehicle Integration



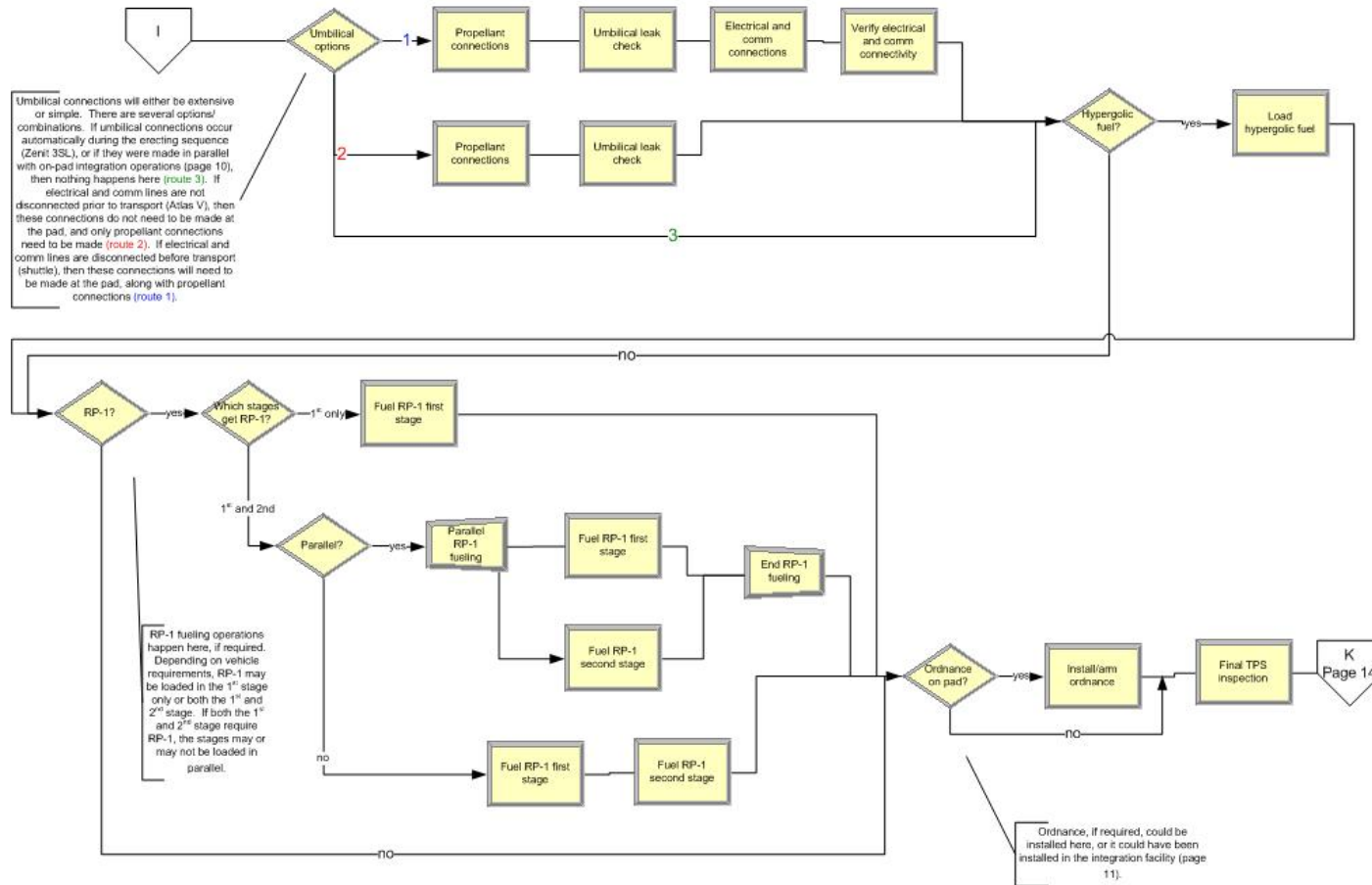
## Round One, Off-pad Vehicle Integration

## Launch Pad Operations for vehicle not integrated on pad



## Round One, Launch Pad Operations for Vehicle not Integrated On Pad

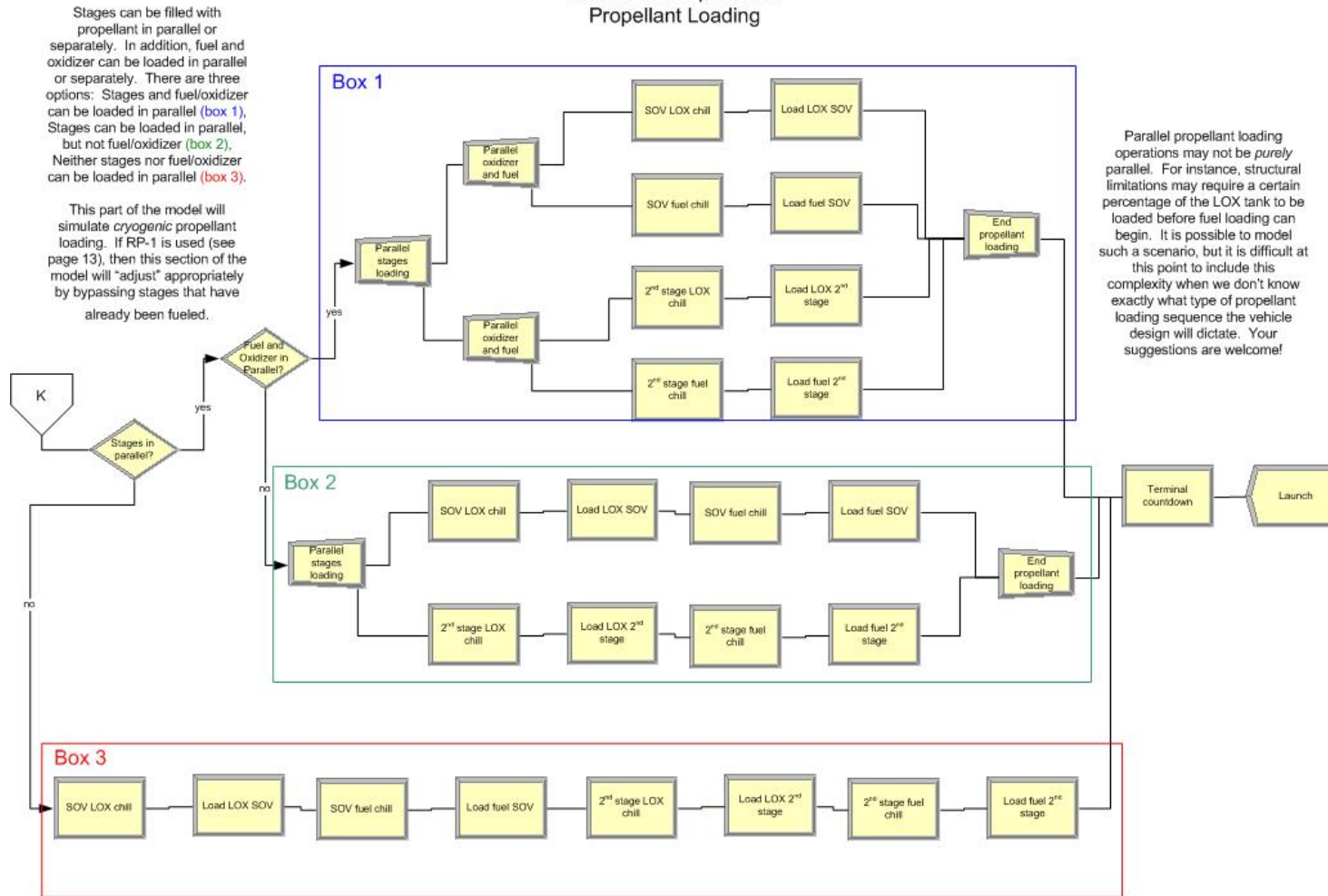
## Launch Pad Operations



13

## Round One, Launch Pad Operations

## Launch Pad Operations Propellant Loading



## Round One, Propellant Loading



## Round One Delphi Member Responses

### PAGE 9

NO COMMENTS

### PAGE 10

**#4** Can the upper stage and payload be mated together while the SOV is going through it's turnaround maintenance then mate the upper stage and payload to the SOV?

**#14** PREP CLEAN ROOM - this is good. Don't you have to transport the vehicle/payload TO the clean room? This could be quite a process. (you also have this option listed on p. 11 and p.12)

**#13** Seems as though some combination of the blocks could happen here as well. Seems to me that the only difference from the pre-integration of the payload path and the no pre-integration path is the fact that the payload is already in the second stage on the upper flow. Would there be a significant change in these distributions for this reason?

**#13** Also, wouldn't there be a 1<sup>st</sup> and 2<sup>nd</sup> stage integration check for the pre-integration flow? I think there may be some differences here since there may be some integration checks to the installed payload if the payload is already on the 2<sup>nd</sup> stage.

**#13** Pre-integration during decision on waiting for launch - Why was the pre-integration process not included on the flow after maintenance inspection? Are you assuming that if the payload is not pre-integrated and the need for launch arises that the payload will "Only" be processed on the No Pre-integration path? This is OK and you mention that the pre-integration takes place to save time (Waiting for launch requirement and assuming the mission ie payload requirement is known). This although precludes the time it takes (Value-Added time?) to pre-integrate the payload into the 2<sup>nd</sup> stage. My thought was that this may need to be included if a launch requirement came before the payload was pre-integrated. Since the launch requirement decision is not modeled you could assume that you know that you have sufficient time to pre-integrate the payload before the decision to launch comes. I know this is long winded and I'm not sure if this/these process should be modeled. I do think you should address these to see if they should be included.

### PAGE 11

**#4** Very busy – can it be broken out? Bottom process – you are loading hypergolics before ordnance install – I believe that the fuel loading should be the last thing done – after ordnance install. Recommend doing all fueling on pad as last and final step – you don't want to move anything with fuels loaded. Also - the pad/payload is purged with nitrogen (inert gas) prior to fuel load - nitrogen atmosphere is unfriendly to humans :-)

**#13** Very thorough but should there be a "Clean Room" on the "Install Payload Now Vertical integration" path?

**#11** Why are pages 10 and 11 so different? All tasks must be completed whether on pad or off...

**PAGE 12**

**#5** The “no” option for “Install payload on pad?” should join the other path before the “entire vehicle integration check” module. Always perform integration check before fueling.

**PAGE 13**

**#4** Umbilical connections. These should be KISS – keep it simple stupid. Connections on the Titan IV were numerous and had connections on almost every stage. These connections get caught on the tower when you move it as well – this I know from experience. Use a race way to carry all connections on the vehicle to one or two concentrated connection points (similar to MMIII) with electrical, fuel, comm, etc. connections.

**PAGE 14**

**#4** Another note – are you going to use squibbed batteries? If you blow the battery and don’t launch then you have to R&R the batteries = lost time.

### **Round Two, Model Changes**

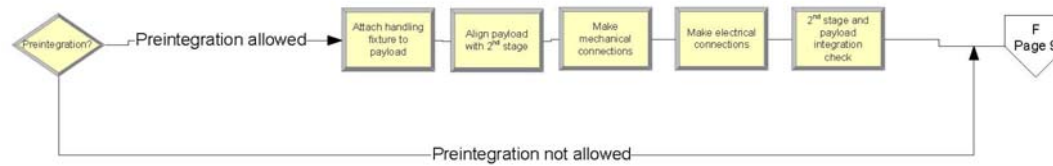
As a result of the comments received in round one, several changes were made to the model in round two. First, a preintegration page was added (diagram on the following page). Second, the “Vehicle Integration—Off-pad Integration” page was changed significantly to make the model easier to follow. In addition, two “storage” options were added to this page to account for an integrated vehicle put into storage if an immediate launch were not necessary.

## Round Two Model Diagram

### Preintegration

Preintegration refers to joining the payload to the second stage before the second stage is attached to the SOV.

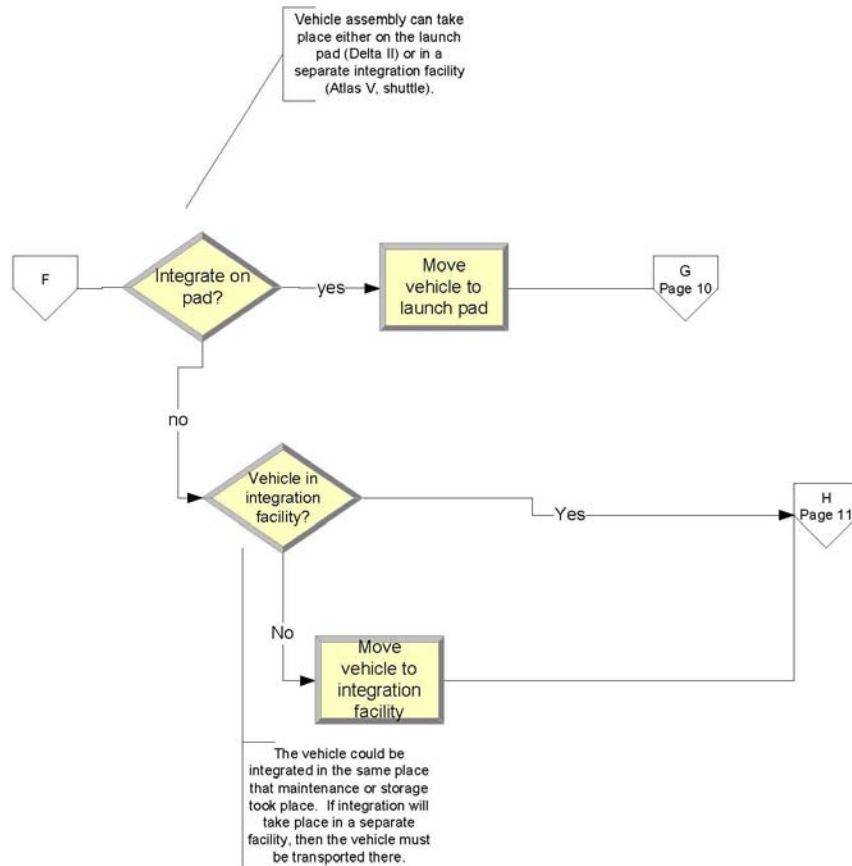
Preintegration assumes that vehicle design and GSE facilitate attaching the second stage/payload to the SOV as a single piece. If vehicle design prohibits this, then payload integration will happen in the traditional fashion, after the second stage is mated to the first stage (see pages 10,11).



NOTE: These events happen in parallel with SOV maintenance (pages 4-8).

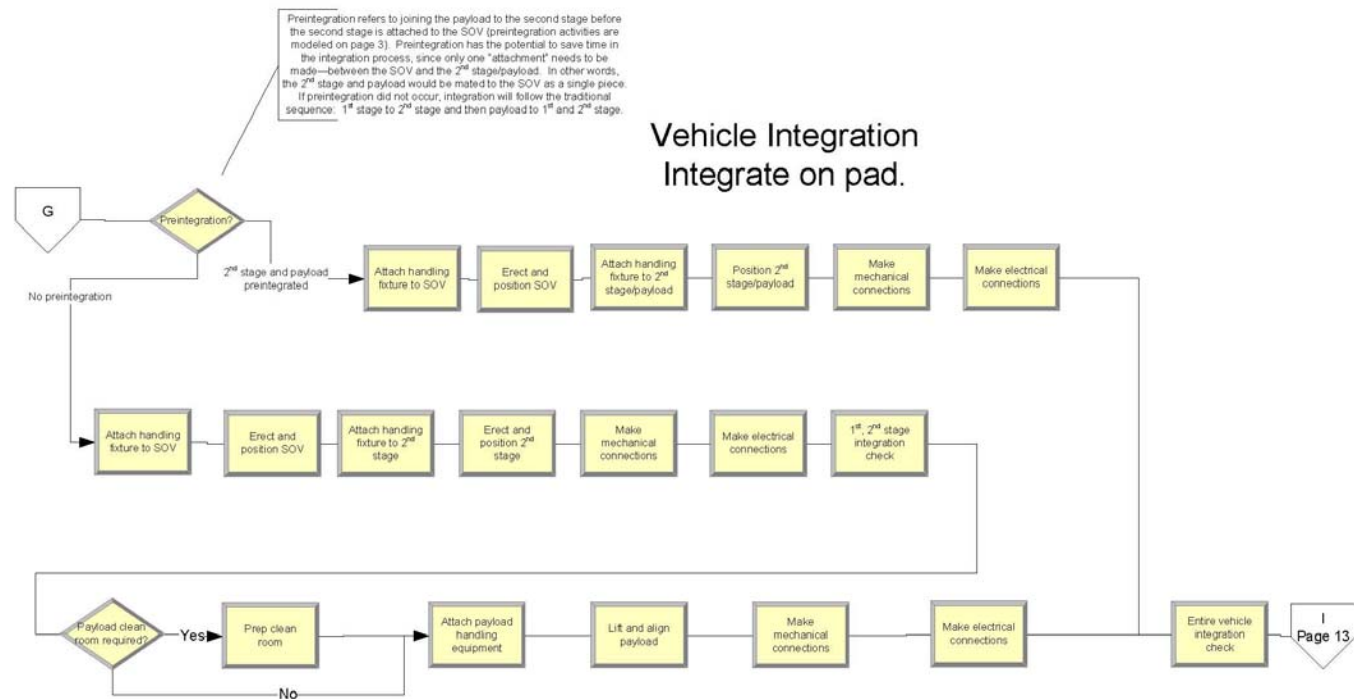
## Round Two, Preintegration

## Vehicle Integration Preliminary Considerations

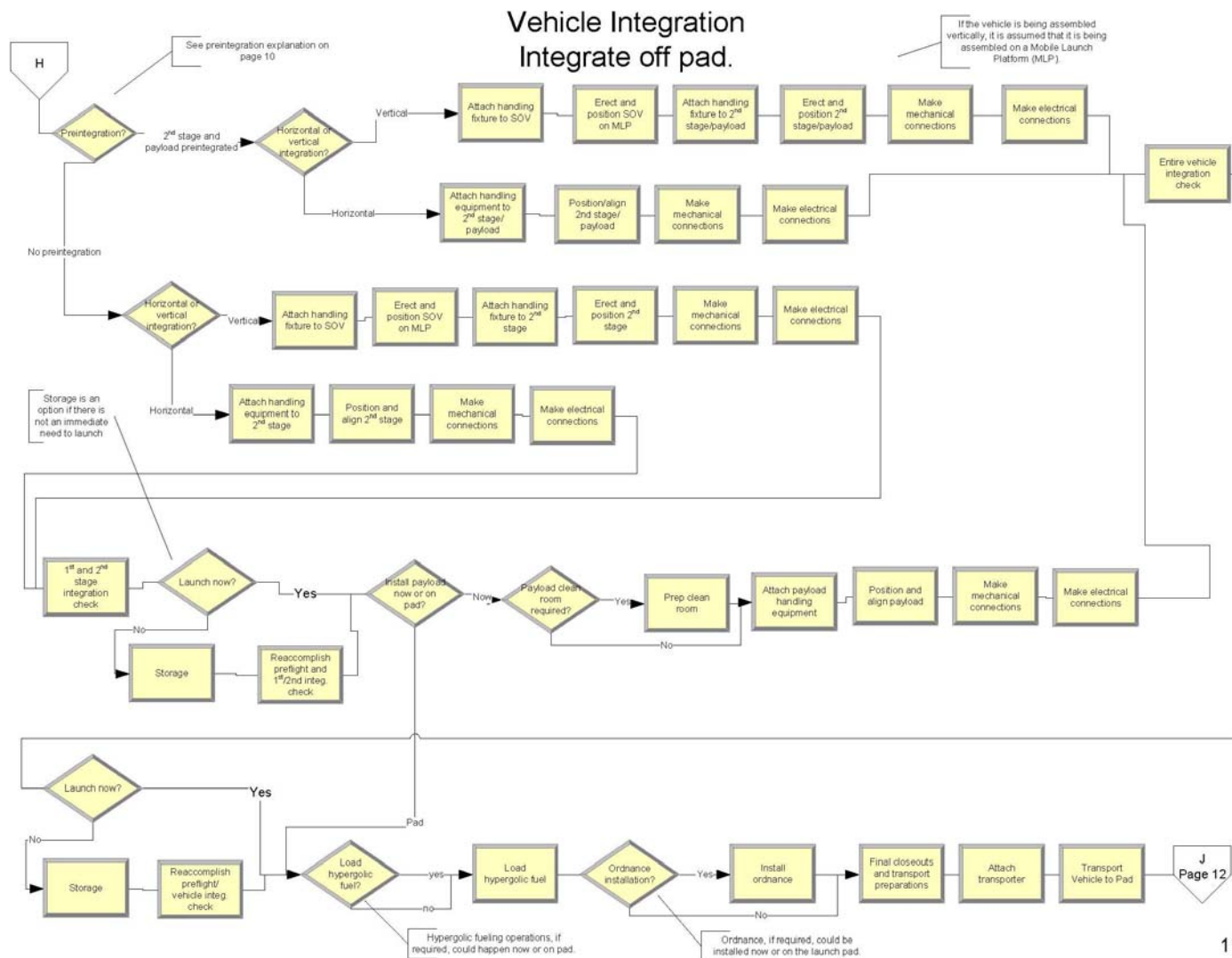


09

### Round Two, Vehicle Integration Preliminary Considerations



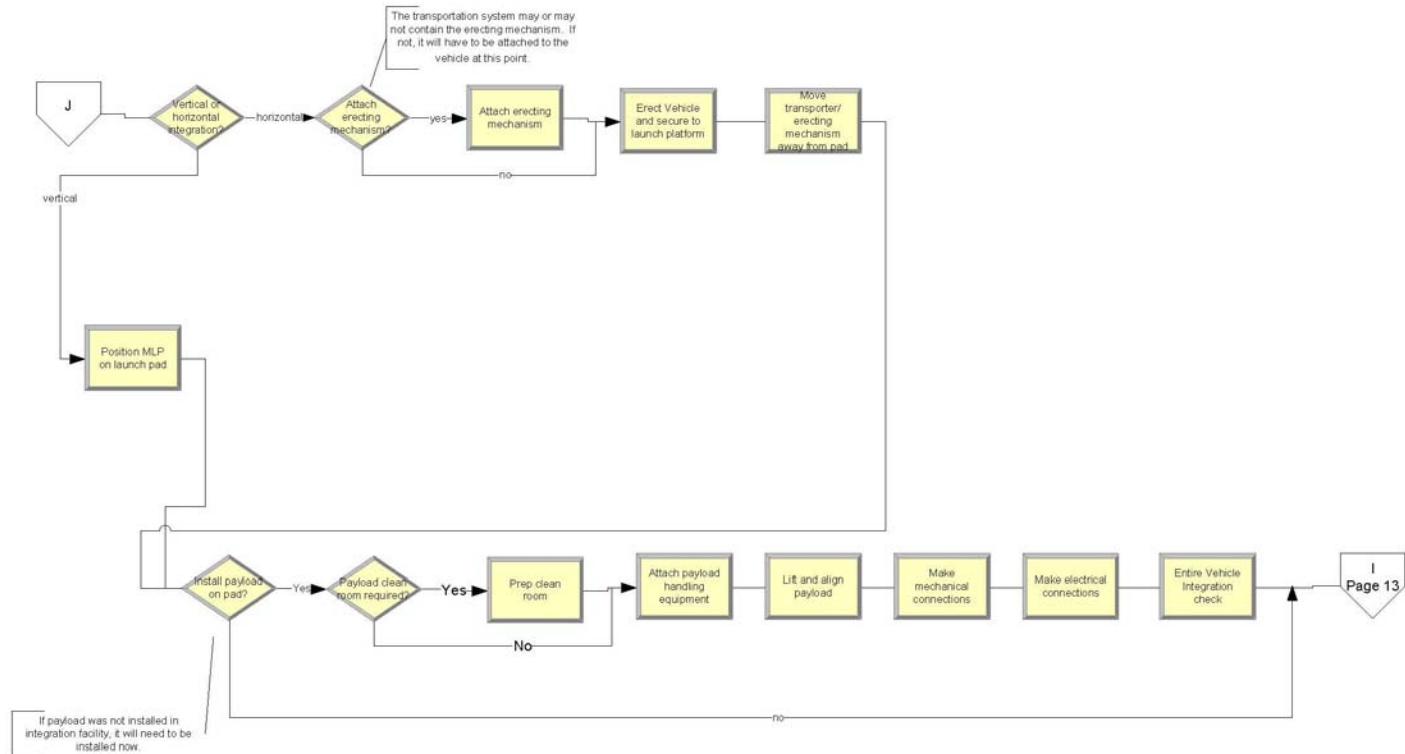
## Round Two, On-pad Vehicle Integration



11

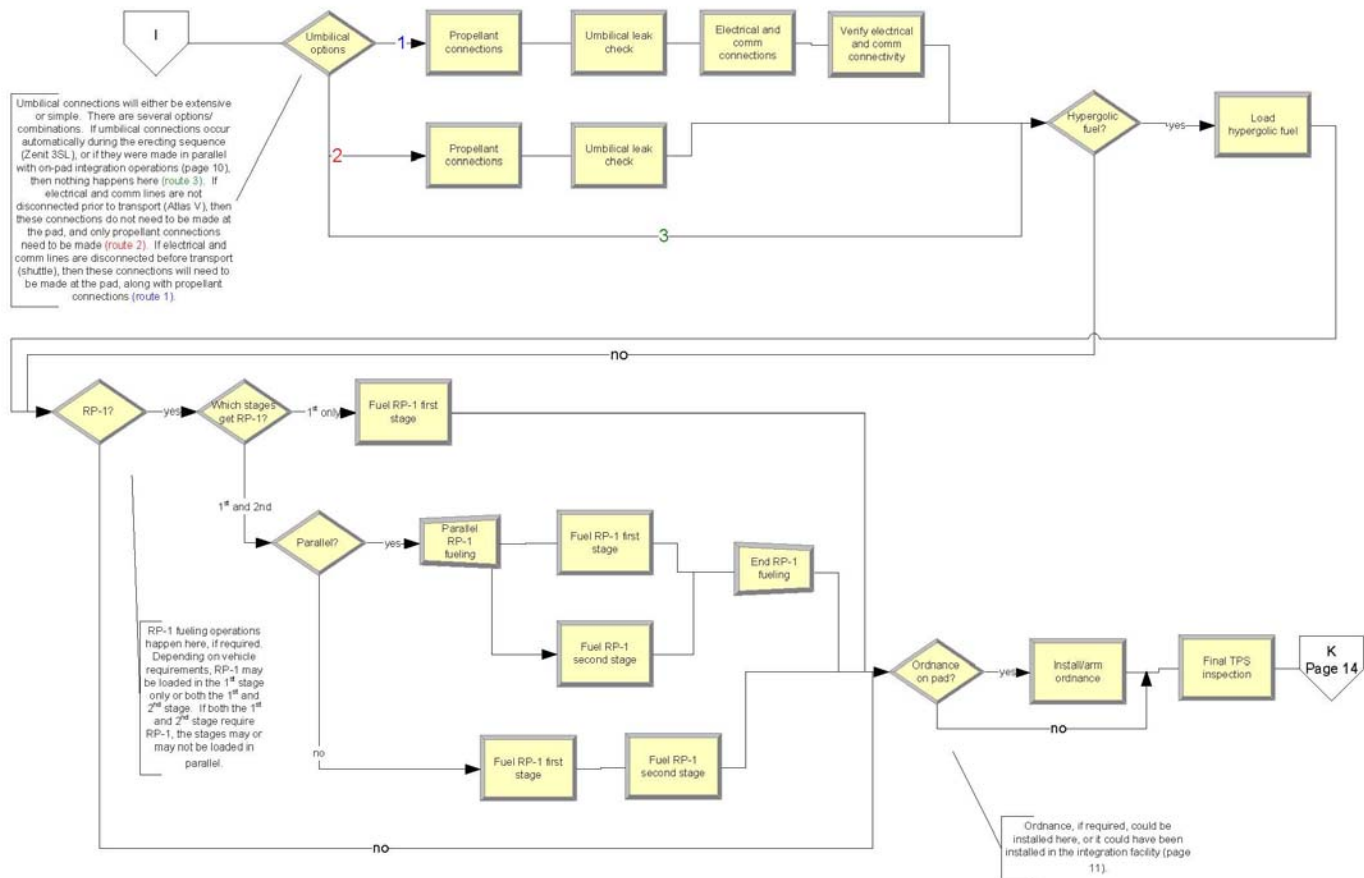
## Round Two, Off-pad Vehicle Integration

## Launch Pad Operations for vehicle not integrated on pad



## Round Two, Launch Pad Operations for Vehicle not Integrated On Pad

## Launch Pad Operations

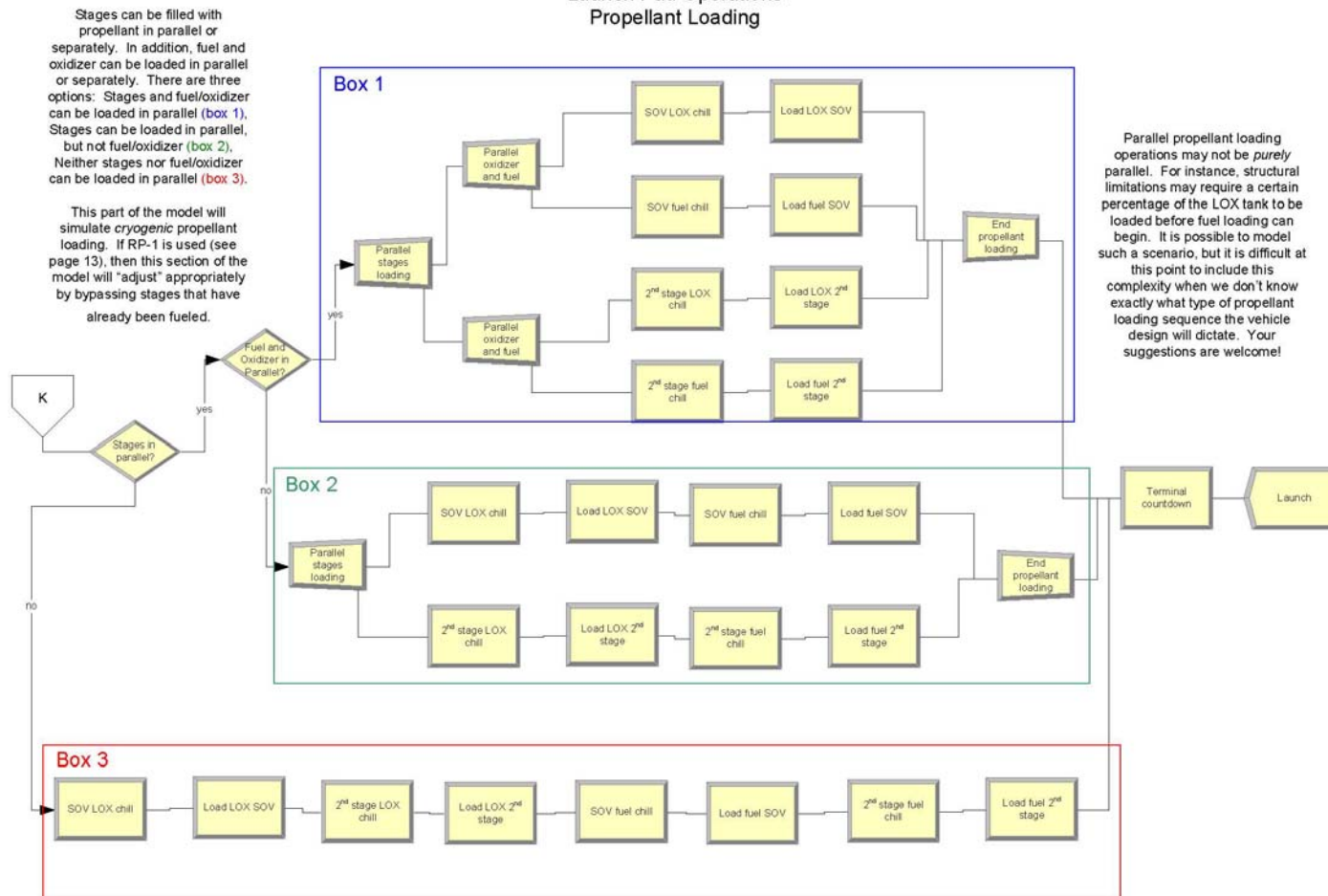


13

## Round Two, Launch Pad Operations



## Launch Pad Operations Propellant Loading



14

## Round Two, Propellant Loading

## Round Two Delphi Member Responses

### PAGE 3

# 13 Will there be a clean room prep requirement for Pre-integration?

### PAGE 9

NO COMMENTS

### PAGE 10

#14 I think you should keep them separate{Two modules that we separated to prevent confusion}. It's possible that a buyer that wants to launch a payload could provide a second stage with the payload already integrated. This option could account for that. Depending on your payload, integrating it on the pad will add the task of moving the payload to the pad and all the logistics that go along with that.

#13 The two separate vehicle erection process flows still seems redundant. If the Pre-integration decision module is moved to before the 1<sup>st</sup>, 2<sup>nd</sup> stage integration check (If this is even needed since it is done again at the vehicle integration check) process, then the affirmative path is directly to the Entire vehicle integration check. The negative path will be to integrate the payload prior to the Entire vehicle integration check.

### PAGE 11

#4 There is usually some type of cooled air supplied to the payload/vehicle while it sits on the pad. It's usually air until the fueling is done and then they switch to nitrogen. The fuel used was cryo so it may be that it is not needed for the hypergolics. The shuttle hypergolics on the shuttle are only a fraction of the total fuel. Also - we transport the PSREs with hypergolics in them but it's a pain.

Transporting a fully fueled vehicle to the pad is a BAD idea - to much risk. What about the added weight of the fuel - will that make your transporter requirements unattainable.

#13 You have a vertical and a horizontal process combining into the 1<sup>st</sup> & 2<sup>nd</sup> stage integration check. Should the down stream process be considered different for payload integration in the vertical or horizontal configuration? These were different paths on the first iteration but now they are combined paths although there was no mention of this in the comments for this page.

Need to extend the "No" path for Load Hypergols Decision module around Load Hypergolic fuel process.

Global Comment on re-inspections. What happens if a problem is found during inspections? Is there information on likelihood of occurrence? Seems that the farther you are in the process the worse the process time might be to fix. Say the TPS is damaged on erecting the completely integrated vehicle on the pad. How this would be repaired might require longer delays than if the damage occurred in the maintenance

facility. This may be beyond the scope of this effort but a processing time hit will result if additional maintenance activities occur.

#### **PAGE 12**

NO COMMENTS

#### **PAGE 13**

**#4** Keep [umbilical connections] as is- I'm probably getting into the weeds on this one :-)

**#14** The connections should be simple, but they are not always simple. I would leave it as is, and if when (if?) we ever build something the tool can be validated with correct times and correct operations. You always have the option of zeroing out time, but I think it would be difficult to take into account an even that did not exist in the model.

**#7** If the Final TPS Inspection done manually it should be performed before the RP-1 fueling to reduce the number of personnel near an already fueled vehicle. If the inspection is automated this would not be required.

#### **PAGE 14**

**#4** If you use hypergolics you can work on the vehicle. If you use cryo - once you fuel then you can't go near it. Plus with all the losses while it sits on the pad you would want to fuel then launch as quick as possible.

I think your only limitation on parallel loading is the infrastructure (pump and pipe size) and the vehicles ability to load multiple tanks at once (structural loading, etc.)

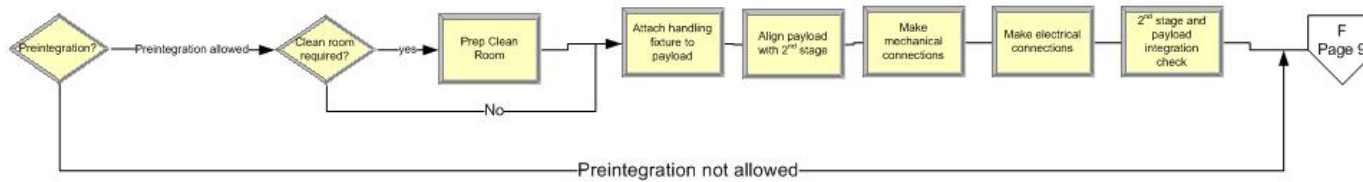
**#14** Is there any way you can create an option to have serial or parallel propellant loading?

### **Round Three, Model Changes**

Only minor changes were made to the model in round three. First, a payload clean room decision was added to the preintegration page. Second, the modules entitled “Reaccomplish preflight and 1<sup>st</sup>/2<sup>nd</sup> stage integ. check” on the “Vehicle Integration—Integrate Off Pad” page were changed to read “Reinspection and additional maintenance” to account for the possibility of a bad inspection requiring additional maintenance.

## Round Three Model Diagram Preintegration

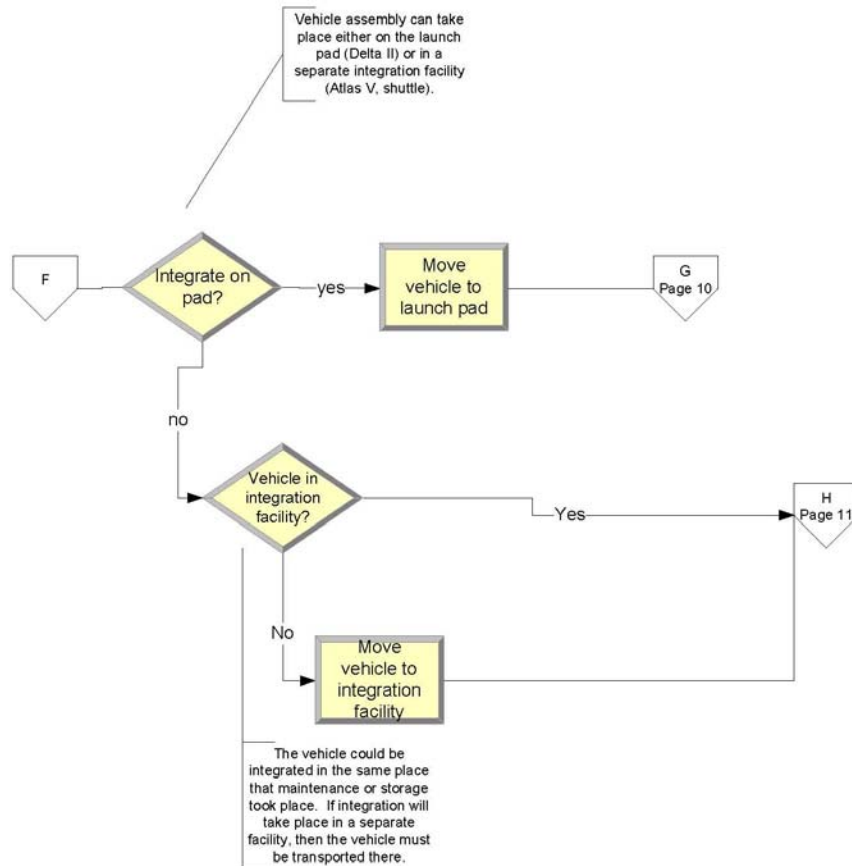
Preintegration refers to joining the payload to the second stage before the second stage is attached to the HLV.  
Preintegration assumes that vehicle design and GSE facilitate attaching the second stage/payload to the HLV as a single piece. If vehicle design prohibits this, then payload integration will happen in the traditional fashion, after the second stage is mated to the first stage (see pages 10,11).



NOTE: These events happen in parallel with HLV maintenance (pages 4-8).

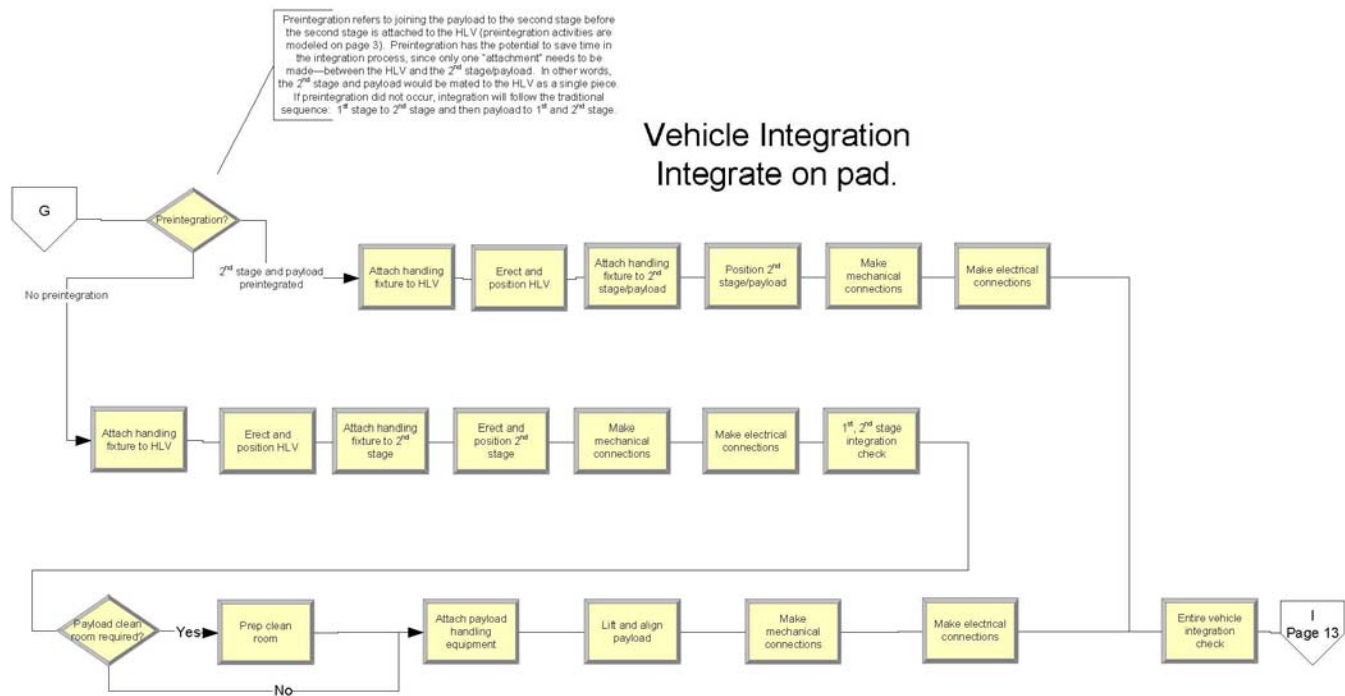
## Round Three, Preintegration

## Vehicle Integration Preliminary Considerations

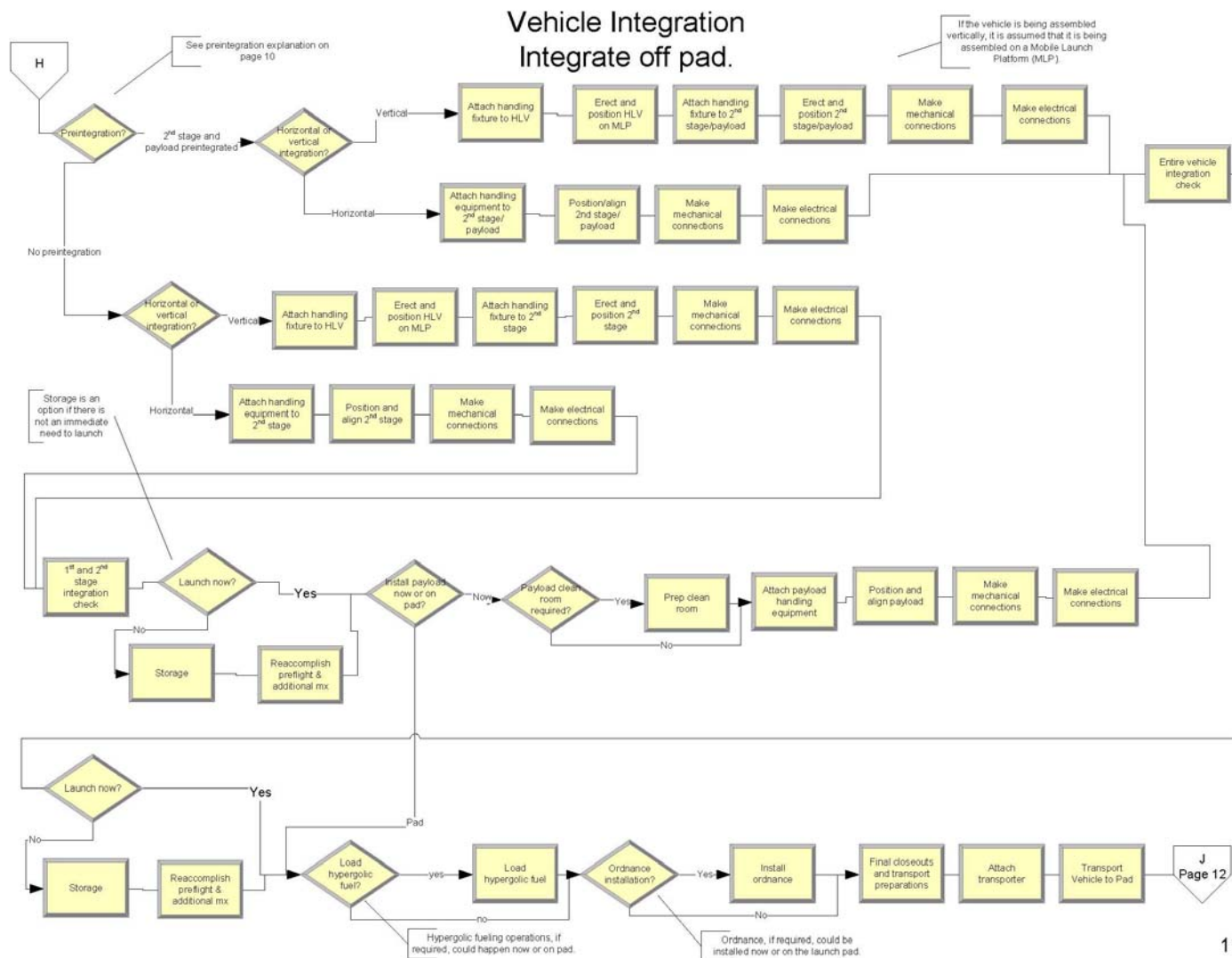


09

**Round Three, Vehicle Integration Preliminary Considerations**



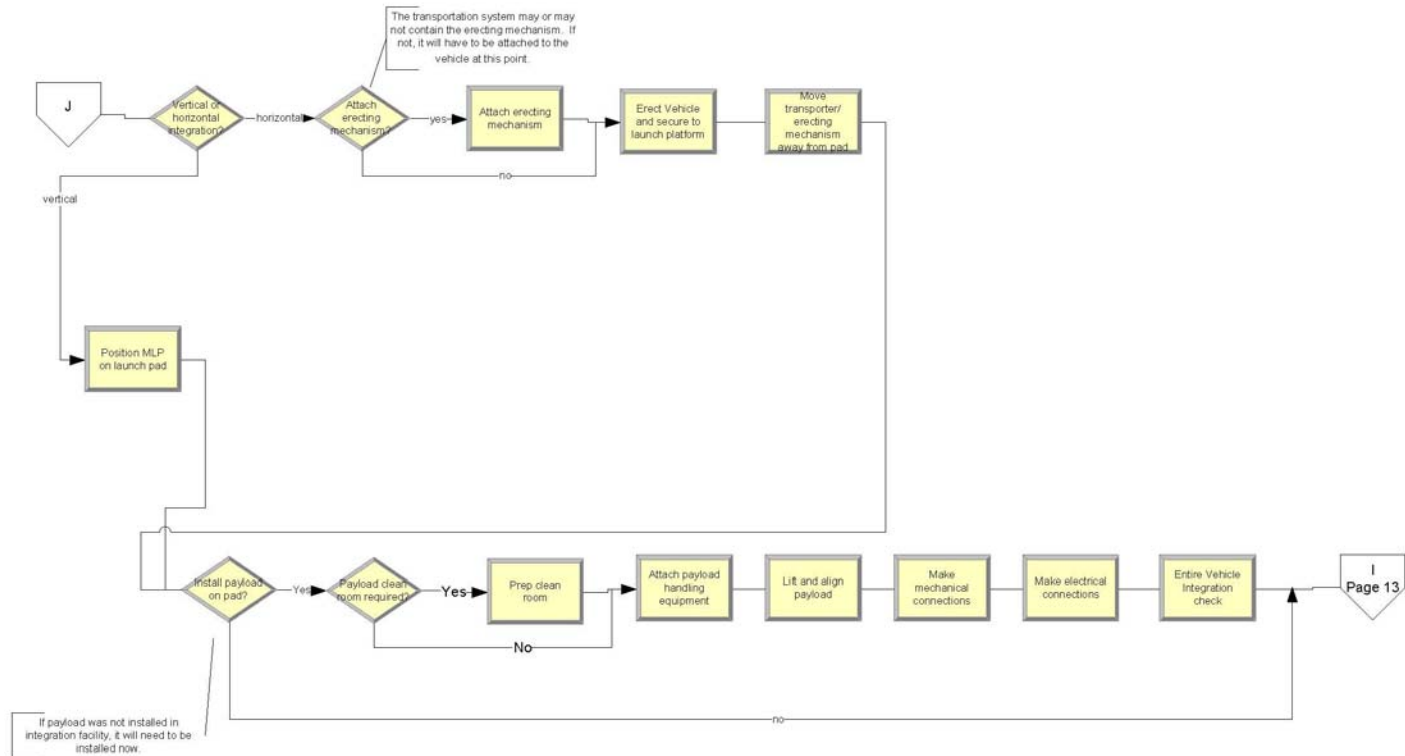
### Round Three, On-pad Vehicle Integration



11

## Round Three, Off-pad Vehicle Integration

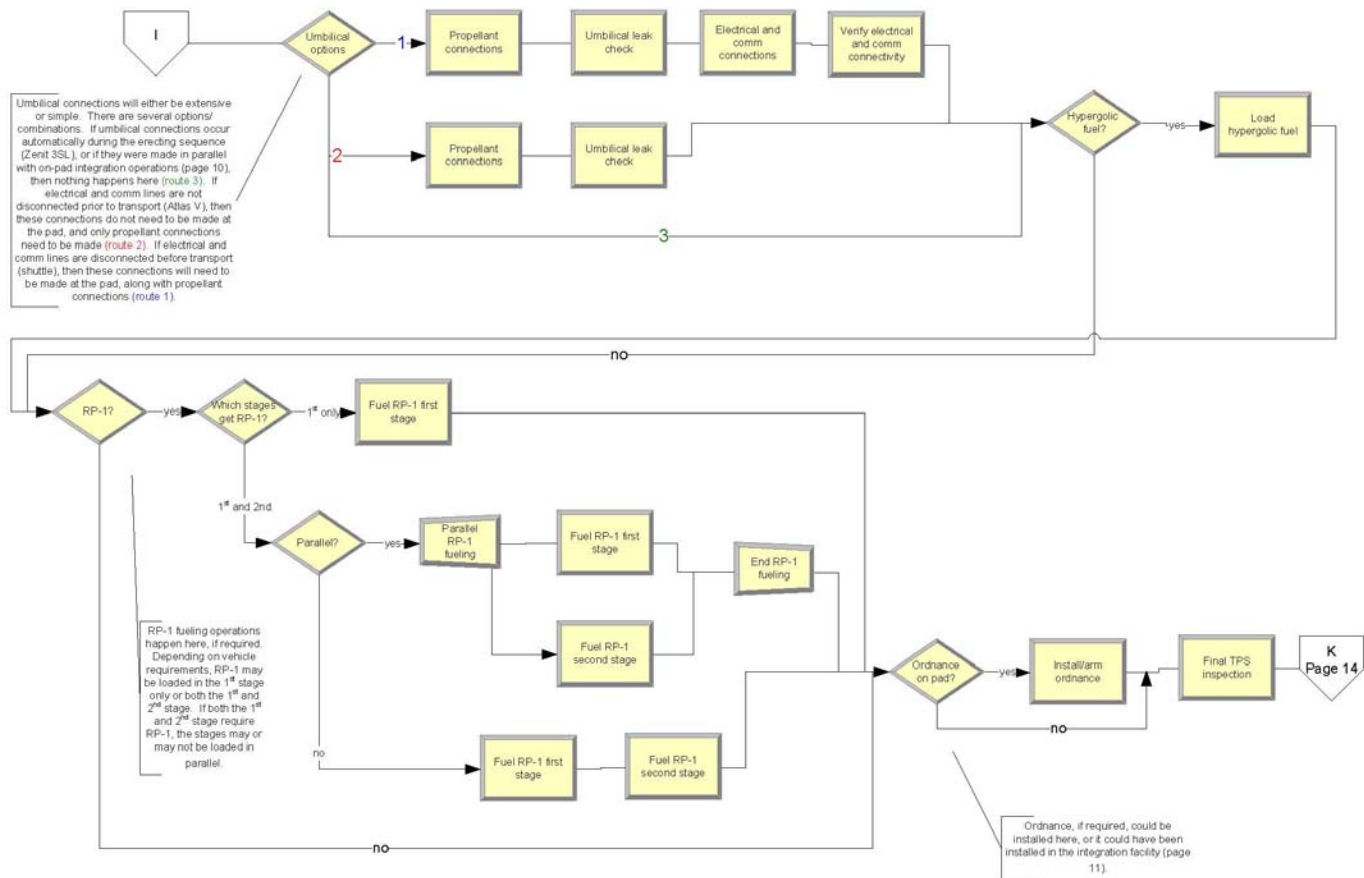
## Launch Pad Operations for vehicle not integrated on pad



## Round Three, Launch Pad Operations for Vehicle not Integrated On Pad



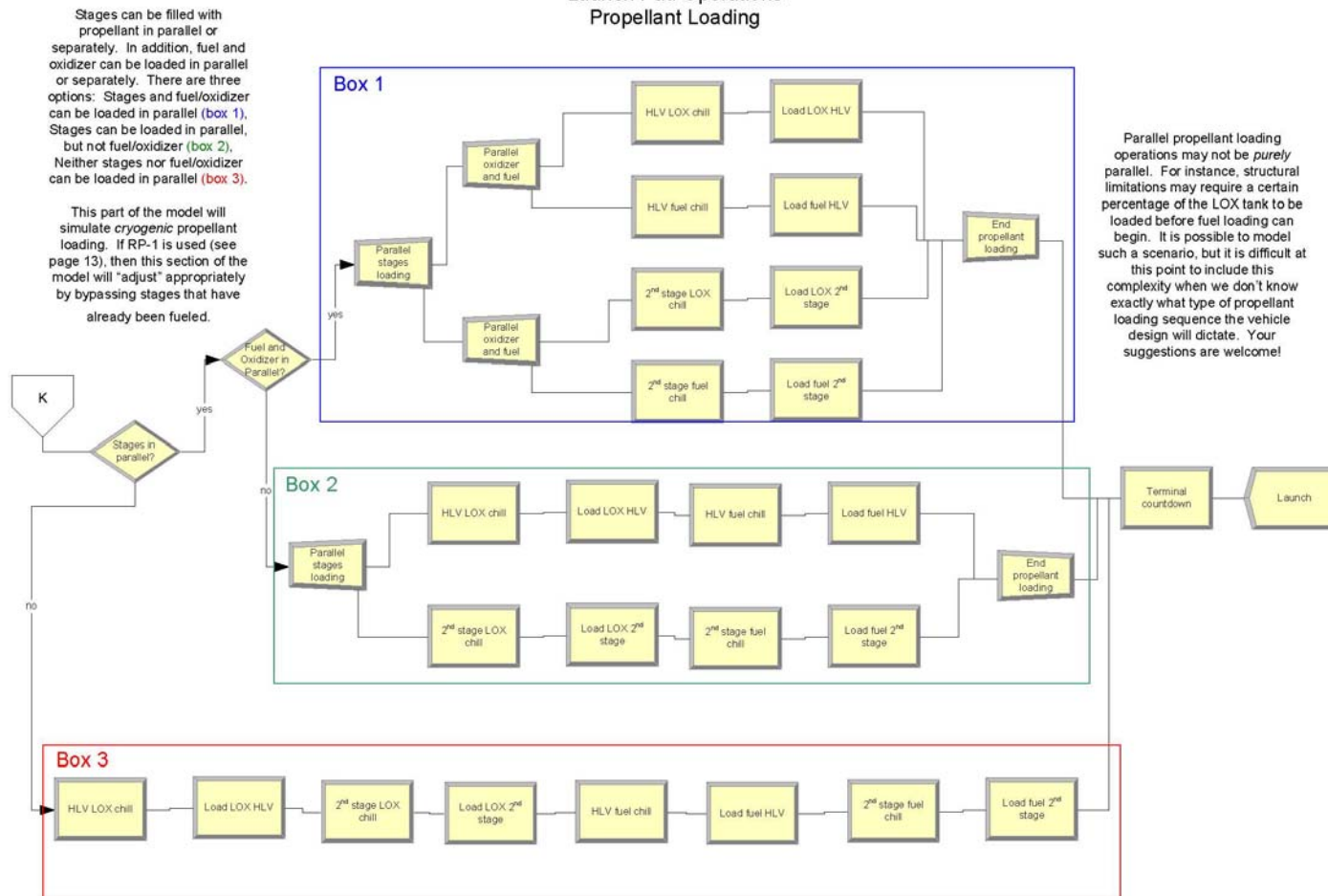
## Launch Pad Operations



13

## Round Three, Launch Pad Operations

## Launch Pad Operations Propellant Loading



14

## Round Three, Propellant Loading

## Round Three Delphi Member Responses

### GENERAL COMMENTS

**#14** I don't mean to be knit-picky here, but I think HLV makes an assumption that may not be true. The first space operating vehicle (SOV) may or may not be a hybrid. It could be, but it also could be fully reusable or completely expendable. I thought SOV was the best way to name the vehicle because it does not assume or imply anything except that the vehicle can operate in space. Operationally Responsive Space Lift/Access (ORS) does not really denote a vehicle, but really describes a *type* vehicle. I would have kept it: SOV. This is probably minor, though.

**#2** From my perspective, no missing events, paths make sense, model appears sound, no recommended changes/deletions/additions/comments.

**#1** I reviewed the model and have no additional input.

**#9** I have no further comments or questions. I enjoyed participating in the development process.

### PAGE 3

**#10** Several instances of using a clean room as a decision block in the payload processing flow diagrams. If the payload is so sensitive to contamination at this stage of the processing, then the flow diagram needs to include encapsulating the payload once your connections have been made and verified. Encapsulated payloads require constant monitoring and a constant supply of clean, dry, regulated air or nitrogen purge. A more realistic approach might be to assume that the payload comes pre-serviced and encapsulated. The Launch team is only responsible for power, comm, and mechanical connections with no clean room required. The EELV payloads usually arrive in this manner, hypergolic propellants are already loaded (but they do have a limited shelf life once they are loaded (30 days??)).

## Model Changes Made After Third Round

As a result of the comments received in round three and modeling decisions made by the author, several changes were made to the final model. These changes are represented in Figures 22-29. First, the payload clean room option was removed completely, from each page where it was used previously. Second, propellant loading chill and fill operations were combined into a single module for each stage and propellant combination (Figure 29). In round three, the chill and fill operations are separate

processes for each stage and propellant combination. The author combined the chill and fill processes because they are closely linked, and, depending on the type of chill procedure chosen, chilling may actually occur as propellant filling begins. Finally, the storage options were removed from the off-pad integration section (Figure 26) because they do not fit with the purpose of the model in its current form, which is to provide a regeneration time for a single RMLV. When the model is modified to incorporate the full spectrum of RMLV operations, the storage options can be reinserted.

## Appendix B: Graphical User Interface User Forms

This Appendix contains the eight graphical user interface forms that were created to simplify user inputs.

**(PI) Preliminary Integration Considerations**

This form gathers user inputs that define whether preintegration is allowed and whether integration of the 1st and 2nd stage will take place on the launch pad or in an integration facility.

Will stage 2 and the payload be preintegrated? Preintegration is defined as the integration of payload and stage 2 before stage 1 is joined to stage 2. A preintegrated stage 2 and payload will be joined to stage 1 as a single piece. In this model, preintegration takes place while stage 1 maintenance is taking place.

Will stage 1 to stage 2 or stage 1 to preintegrated stage 2/payload integration take place on the launch pad or off the launch pad?

PI-01 ☒ Preintegration ☐ No preintegration

PI-02 Attach handling fixture to payload TRIA ( 27, 30, 42 ) Minutes

PI-03 Align the payload with stage 2 TRIA ( 27, 30, 42 ) Minutes

PI-04 Stage 2 to payload mechanical connections TRIA ( 18, 20, 28 ) Minutes

PI-05 Stage 2 to payload electrical connections TRIA ( 18, 20, 28 ) Minutes

PI-06 Stage 2 to payload integration check TRIA ( 27, 30, 42 ) Minutes

PI-07 ☐ launch pad ☐ off launch pad

PI-08 For on-pad integration, time to transport stage 1 from maintenance bay to the launch pad TRIA ( 27, 30, 42 ) Minutes

If integration will happen off the launch pad, will it occur where Stage 1 maintenance took place (maintenance bay), or will it occur in a separate integration facility?

PI-09 ☐ maintenance bay ☐ integration facility

PI-10 For off-pad integration, time to transport Stage 1 from maintenance bay to integration facility TRIA ( 13.5, 15, 21 ) Minutes

Previous Next

Main Help

### Preliminary Integration Considerations

## On-pad Integration







(OT) Other tasks, vehicle integration facility

This form gathers user inputs that describe final tasks within the integration facility, such as hypergolic fuel loading, ordnance install, and vehicle transport preparations.

Will hypergolic fuels be required?

OT-01
☐ hypergolic fuels required
☐ Hypergolic fuels not required

Click on the appropriate location for hypergolic fueling. Hypergolic fuels can be loaded now, in the integration facility, or later, on the launch pad.

OT-02
☐ Load hypers in integration facility
☐ Load hypers on launch pad

OT-03 Time to load hypergolic fuels

TRIA ( 756, 840, 1176 )

Minutes

OT-07 Prepare for vehicle transport to pad

TRIA ( 108, 120, 168 )

Minutes

OT-08 Attach vehicle transporter to the vehicle

TRIA ( 9, 10, 14 )

Minutes

OT-09 Transport vehicle to launch pad

TRIA ( 27, 30, 42 )

Minutes

Will ordnance be required?

OT-04
☐ Ordnance required
☐ Ordnance not required

Click on the appropriate location for ordnance installation. Ordnance can be installed now, in the integration facility, or later, on the launch pad.

OT-05
☐ Install ordnance in integration facility
☐ Install ordnance on launch pad

OT-06 Time to install ordnance

TRIA ( 324, 360, 504 )

Minutes

Previous
Next

Main
Help

## Other Tasks, Vehicle Integration Facility





(PL) Propellant loading

This form gathers user inputs that describe cryogenic propellant loading operations.

Choose one of the following three cryogenic propellant loading options.

☐

STRICT SERIAL PROPELLANT LOADING. Loading of fuel and oxidizer and loading of stages cannot happen in parallel, i.e. LOX cannot be loaded into the 1st and 2nd stage at the same time, and LOX and LH2 cannot be loaded into the 1st stage at the same time.

☐

PL-01 PARALLEL STAGE LOADING, SERIAL PROPELLANT LOADING. The 1st and 2nd stage may be loaded in parallel, but fuel and oxidizer cannot be loaded in parallel, i.e. LOX may be loaded into the 1st and 2nd stage at the same time, but LOX and LH2 may not be loaded at the same time.

☐

PARALLEL STAGE AND PROPELLANT LOADING. Stages and fuel/oxidizer may be loaded in parallel, i.e. both LOX and LH2 can be loaded into the 1st and 2nd stage all at once.

PL-02 Chill and fill 1st stage LOX

TRIA ( 54, 60, 84 )

Minutes

PL-03 Chill and fill 2nd stage LOX

TRIA ( 27, 30, 42 )

Minutes

PL-04 Chill and fill 1st stage fuel

TRIA ( 54, 60, 84 )

Minutes

PL-05 Chill and fill 2nd stage fuel

TRIA ( 27, 30, 42 )

Minutes

PL-06 Length of terminal countdown

10

Minutes

Previous

Main

Help

## Propellant Loading

## Appendix C: Prelaunch Operations Activity Duration Distributions

<u>All times are in minutes</u>				
<u>Activity</u>	<u>Min</u>	<u>Mode</u>	<u>Max</u>	<u>Source</u>
<b>Payload Vertical</b>				
Lift and align pld w/ 2nd stage (vertical)	81	90	126	Atlas tour, Delta II planners guide
2nd stage to pld mech connections (vertical)	27	30	42	Minuteman III
2nd stage to pld electric connections (vertical)	27	30	42	Minuteman III
<b>Payload Horizontal</b>				
Align payload w/2nd stage (horizontal)	27	30	42	AFRL/author's aircraft maintenance experience
2nd stage to pld mech connections (horizontal)	18	20	28	Minuteman III minus 33%
2nd stage to pld electric connections (horizontal)	18	20	28	Minuteman III minus 33%
<b>Payload Generic</b>				
Attach handling fixture to payload	27	30	42	Minuteman III
<b>Transport</b>				
Move from mx bay to launch pad	27	30	42	Atlas tour, Atlas moves at top speed of 2 mph
Move from mx bay to int facility	13.5	15	21	Atlas tour, Atlas moves at top speed of 2 mph
Move from int facility to launch pad	27	30	42	Atlas tour, Atlas moves at top speed of 2 mph
Transport Preparations	108	120	168	Atlas/author's aircraft mx experience
Attach transporter	9	10	14	Aircraft (U-2) tow
<b>Integration Vertical</b>				
Lift and position 1st stage on pad or MLP	108	120	168	Shuttle/Delta II planners guide
Lift and align 2nd stage or 2nd stage/payload	81	90	126	Atlas tour, Delta II planners guide
1st to 2nd mechanical connections vert	36	40	56	Horizontal value plus 33% penalty
1st to 2nd electrical connections vert	36	40	56	Horizontal value plus 33% penalty
<b>Integration Horizontal</b>				
Position and align 2nd stage or 2nd stage/payload	54	60	84	AFRL/author's aircraft maintenance experience
1st to 2nd mechanical connections horiz	27	30	42	AFRL/author's aircraft maintenance experience
1st to 2nd electrical connections horiz	27	30	42	AFRL/author's aircraft maintenance experience
<b>Integration generic</b>				
Integration check	27	30	42	AFRL/author's aircraft maintenance experience
Attach handling fixture to booster	54	60	84	Shuttle
Attach handling fixture to 2nd stage	27	30	42	Minuteman III
<b>Initial Launch Pad Positioning</b>				
Secure MLP to launch platform	54	60	84	Atlas
Attach erecting mechanism to vehicle	27	30	42	AFRL/author's aircraft maintenance experience
Erect vehicle	27	30	42	Sea Launch
<b>Misc</b>				
Install ordnance	324	360	504	Atlas
Load hypergolic fuel	756	840	1176	Shuttle
Final TPS or other inspection	0	0	0	Uncertain--depends on vehicle requirements
Terminal countdown	10	10	10	Atlas uses 4 minute terminal count, plus other holds
<b>RP</b>				
Fill RP 1st stage	108	120	168	Atlas
Fill RP 2nd stage	54	60	84	Atlas
<b>Umbilicals</b>				
Make propellant connections	27	30	42	Atlas
Leak check propellant connections	4.5	5	7	Atlas
Make electrical/comm connections	27	30	42	AFRL/author's aircraft maintenance experience
Verify electrical/comm connections	27	30	42	AFRL/author's aircraft maintenance experience
<b>Cryos</b>				
1st stage LOX chill and fill	54	60	84	EELV/Shuttle
1st stage fuel chill and fill	54	60	84	EELV/Shuttle
2nd stage LOX chill and fill	27	30	42	EELV/Shuttle
2nd stage fuel chill and fill	27	30	42	EELV/Shuttle

## Bibliography

- Andrews, Julie. Lockheed Martin Communications, Cape Canaveral, FL. Personal Correspondence. 31 January 2006.
- Atlas Launch System Mission Planner's Guide*. International Launch Services, Inc. and Lockheed Martin Corporation, McLean, VA, 2004. [http://ilslaunch.com/whatsnew/Publications\\_Advertising/](http://ilslaunch.com/whatsnew/Publications_Advertising/). 26 July 2005.
- Banks, J., Carson, J.S. II, Nelson, B.L., Nicol, D.M., *Discrete-Event System Simulation*, Upper Saddle River, NJ: Pearson Prentice Hall, 2005.
- Brown, Kendall K. "A Concept of Operations and Technology Implications for Operationally Responsive Space," *Air & Space Power Chronicles, Chronicles Online Journal*, 28 July 2004. <http://www.airpower.maxwell.af.mil/airchronicles/cc/brown2.html>. 26 May 2005.
- Brown, Kendall K. *Technology Challenges for Operationally Responsive Spacelift* (Airpower Research Institute Paper No. 2004-02). Maxwell AFB, AL: Airpower Research Institute, 2004.
- Cates, G.R., Steele, M.J., Mollaghasemi, M., Rabadi, G., "Modeling the Space Shuttle," *Proceedings of the 2002 Winter Simulation Conference*, pp. 754-762.
- Cates, Grant, Shuttle Processing Operations Integration Office, Kennedy Space Center. "Space Shuttle Processing and Metrics Overview." Powerpoint Slideshow. 26 June 2003.
- Cates, Grant, Shuttle Processing Operations Integration Office, Kennedy Space Center. "The Manifest Assessment Simulation Tool (MAST)." Briefing presented to author, Kennedy Space Center, FL, 17 August 2005.
- Centore, Jim, Lockheed Martin Corporation. "Atlas Launch Facilities and Launch Processing Operations," Briefing presented to author, Kennedy Space Center, FL, 18 August 2005.
- "The Delphi Method." n. pag. <http://www.iit.edu/~it/delphi.html>. 13 September 2005
- "Delta IV Launch Vehicle and Spacecraft Processing." n. pag. [http://www.boeing.com/defense-space/space/bls/d4heavy/d4h\\_process.html](http://www.boeing.com/defense-space/space/bls/d4heavy/d4h_process.html). 8 June 2005.
- Delta Payload Planner's Guide*. The Boeing Company, Huntington Beach, CA, 2000. <http://www.boeing.com/defense-space/space/delta/guides.htm>. 26 July 2005.
- Defense Modeling and Simulation Office. *Verification, Validation, and Accreditation*

*Recommended Practices Guide*. November 1996.

Dornheim, Michael A. "Quick, Cheap Launch: Year-long USAF Study Aims for New Launchers by 2014," *Aviation Week and Space Technology*, 158: 70-72 (7 April 2003).

Dornheim, Michael A. "A Quick Response," *Aviation Week and Space Technology*, 162: 33-34 (2 May 2005).

Ebeling, Charles E. *An Introduction to Reliability and Maintainability Engineering*, Long Grove, IL: Waveland Press, 2005.

"Evolved Expendable Launch Vehicle," U.S. Air Force Fact Sheet, May 2005. n. pag. [http://www.af.mil/factsheets/factsheet\\_print.asp?fsID=169&page=1](http://www.af.mil/factsheets/factsheet_print.asp?fsID=169&page=1). 1 August 2005.

General Accounting Office. *Defense Space Activities: Continuation of Evolved Expendable Launch Vehicle Program's Progress to Date Subject to Some Uncertainty*. GAO-04-778R. 24 June 2004.

Greaves, Samuel A. *The Evolved Expendable Launch Vehicle (EELV) Acquisition and Combat Capability*. Research Paper, Air Command and Staff College, Air University, Maxwell AFB, AL, March 1997. <http://www.globalsecurity.org/space/library/report/1997/97-0133.pdf>.

"Hybrid Launch Vehicle." U.S. Air Force Fact Sheet. <http://www.losangeles.af.mil/SMC/XR/PUBLIC/bidderslibrary/documents/hlvfactsheet.doc>. 13 September 2005.

"Hybrid Launch Vehicle Studies and Analysis Statement of Objectives (HLV S&A SOO)." <http://www.losangeles.af.mil/SMC/XR/PUBLIC/bidderslibrary/documents/hlvsasoo.doc>. 13 September 2005.

*ICBM Familiarization*. Missile Maintenance Officer Training Block I Study Guide. 532d Training Squadron, Vandenberg AFB, CA, 18 April 2001.

Isakowitz, Steven J., Hopkins, Joseph P. Jr, Hopkins, Joshua B. *International Reference Guide to Space Launch Systems* (3<sup>rd</sup> Edition), Reston, VA: American Institute of Aeronautics and Astronautics, 1999.

Isakowitz, Steven J., Hopkins, Joshua B., Hopkins, Joseph P. Jr. *International Reference Guide to Space Launch Systems* (4<sup>th</sup> Edition), Reston, VA: American Institute of Aeronautics and Astronautics, 2004.

"LGM 30 Minuteman III." n. pag. [http://www.fas.org/nuke/guide/usa/icbm/lgm-30\\_3.htm](http://www.fas.org/nuke/guide/usa/icbm/lgm-30_3.htm). 14 September 2005.

- McClave, James T., Benson, P. George, Sincich, Terry. *Statistics for Business and Economics*, Upper Saddle River, NJ: Pearson Prentice Hall, 2005.
- McCleskey, Carey M. *Space Shuttle Operations and Infrastructure: A Systems Analysis of Design Root Causes and Effects*, NASA Scientific and Technical Information Program Office Technical Report, NASA/TP—2005—211519. April 2005.
- “NASA’s Exploration Systems Architecture Study,” Executive Summary, November 2005. [http://www.nasa.gov/pdf/140631main\\_ESAS\\_01.pdf](http://www.nasa.gov/pdf/140631main_ESAS_01.pdf). 13 February 2006.
- Neufeld, Jacob. *The Development of Ballistic Missiles in the United States Air Force, 1945-1960*. Washington, D.C.: Office of Air Force History, United States Air Force, 1990. 24 February 2006. [https://www.airforcehistory.hq.af.mil/Publications/fulltext/ballistic\\_missiles\\_in\\_the\\_usaf.pdf](https://www.airforcehistory.hq.af.mil/Publications/fulltext/ballistic_missiles_in_the_usaf.pdf)
- O’Malley, Timothy. B-2 Systems Group, Long Range Strike Wing, Aeronautical Systems Center, Wright-Patterson Air Force Base, OH. Personal Correspondence. 27 January 2006.
- “Pad SSV TA Days.” Microsoft Excel data file. Courtesy of Kennedy Space Center, FL, no date.
- Pope, John T. III. *Discrete Event Simulation Model of the Ground Maintenance Operations of a Reusable Military Launch Vehicle*. MS Thesis, AFIT/GLM/ENS/06-14. Graduate School of Engineering and Management, Air Force Institute of Technology (AU), Wright Patterson Air Force Base, Ohio, March 2006.
- Pope, John T. III. Missile Maintenance Officer, United States Air Force. Personal Interview. 14 September 2005.
- “President Bush Announces New Vision for Space Exploration Program.” White House Press Release, 14 January 2004. n. pag. <http://www.whitehouse.gov/news/releases/2004/01/print/20040114-1.html>. 4 August 2005.
- Rooney, Brendan D. and Hartong, Alicia. “A Discrete-Event Simulation of Turnaround Time and Manpower of Military RLVs,” *A Collection of Technical Papers—AIAA Space 2004 Conference and Exposition*. 2236-2253. Reston, VA: American Institute of Aeronautics and Astronautics, 2004.
- “Sea Launch Image Gallery.” The Sea Launch Company. n. pag. <http://www.sea-launch.com/images.htm>. 15 September 2005.
- Steele, M.J., Mollaghasemi, M., Rabadi, G., Cates, G. “Generic Simulation Models of Reusable Launch Vehicles.” *Proceedings of the 2002 Winter Simulation Conference*, pp. 747-752.



“TA Days.” Microsoft Excel data file. Courtesy of Kennedy Space Center, FL, no date.

Turoff, Murray and Hiltz, Starr Roxanne. “Computer Based Delphi Processes.” A version of this paper also appears as an invited book chapter in *Gazing Into the Oracle: The Delphi Method and Its Application to Social Policy and Public Health*, Michael Adler and Erio Ziglio, editors. <http://eies.njit.edu/~turoff/Papers/delphi3.html>. 13 September 2005.

“Vehicle Assembly Building.” Kennedy Space Center, 1999. n. pag. <http://science.ksc.nasa.gov/facilities/vab.html>. 13 February 2006.

Wall, Robert. “USAF, NASA Eye Cooperation on Next Reusable Launcher,” *Aviation Week and Space Technology*, 156:34-39. 28 January 2002.

Wilson, J. R., Vaughan, D.K., Naylor, E., Voss, R.G. “Analysis of Space Shuttle Ground Operations,” *Simulation*, June 1982, pp. 187-203.

## **Vita**

Captain Adam T. Stiegelmeier graduated high school from Sunshine Bible Academy in Miller, SD. He earned his bachelor's degree in Political Science from Cedarville University in Cedarville, OH. He was commissioned out of AFROTC Detachment 643 at Wright State University.

Captain Stiegelmeier's first assignment was at Beale AFB, where he served as an aircraft maintenance officer in both the 9<sup>th</sup> Maintenance Squadron and the 9<sup>th</sup> Aircraft Maintenance Squadron. While at Beale, he deployed to Prince Sultan Air Base for five months in support of Operation Iraqi Freedom. In August of 2004, he was assigned to the Air Force Institute of Technology where he earned his master's degree in Logistics Management. Upon graduation he will be assigned to Headquarters Air Mobility Command, Scott AFB, IL.

REPORT DOCUMENTATION PAGE				Form Approved OMB No. 074-0188	
<p>The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of the collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.</p> <p><b>PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.</b></p>					
1. REPORT DATE (DD-MM-YYYY) 23-03-2006		2. REPORT TYPE Master's Thesis		3. DATES COVERED (From – To) Mar 2005 – Mar 2006	
4. TITLE AND SUBTITLE  A DISCRETE EVENT SIMULATION MODEL FOR EVALUATING AIR FORCE REUSABLE MILITARY LAUNCH VEHICLE PRELAUNCH OPERATIONS				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S)  Stiegelmeier, Adam, T., Captain, USAF				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S) Air Force Institute of Technology Graduate School of Engineering and Management (AFIT/EN) 2950 Hobson Street, Building 642 WPAFB OH 45433-7765				8. PERFORMING ORGANIZATION REPORT NUMBER  AFIT/GLM/ENS/06-16	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) AFRL/VAOT Attn: Mr. Bruce Thieman 2130 8 <sup>th</sup> Street WPAFB OH 45433 DSN: 785-5162 e-mail: bruce.thieman@wpafb.af.mil				10. SPONSOR/MONITOR'S ACRONYM(S)	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT  APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.					
13. SUPPLEMENTARY NOTES					
14. ABSTRACT <p>As the control and exploitation of space becomes more important to the United States military, a responsive spacelift capability will become essential. Responsive spacelift could be defined as the ability to launch a vehicle within hours or days from the time a launch order is given, instead of the weeks or months it takes currently. As the Air Force contemplates moving toward a reusable military launch vehicle (RMLV) capability, it faces key design and ground processing decisions that will affect the vehicle regeneration timeline. This thesis develops a computer simulation model that mimics RMLV prelaunch operations—those activities that take place during vehicle integration and launch pad operations. This simulation model can help the Air Force make RMLV acquisition decisions by analyzing how different RMLV designs and ground processing scenarios will affect RMLV regeneration time.</p> <p>The model was developed by comparing and contrasting existing launch vehicle processing flows to create the RMLV prelaunch operations model. To foster confidence in model credibility, the model was analyzed and validated by a panel of launch vehicle experts. Model verification was accomplished via an Assertion Checking method that compared model developer intent to actual model operation. The model was used to conduct three experiments that analyzed how different ground processing scenarios affected RMLV regeneration time.</p>					
15. SUBJECT TERMS Reusable Military Launch Vehicle, Reusable Launch Vehicle, Launch Vehicle, Ground Processing, Prelaunch Operations, Integration Operations, Launch Pad Operations, Computer Simulation, Discrete Event Simulation					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Alan W. Johnson, Ph. D.
U	U	U	UU	147	19b. TELEPHONE NUMBER (Include area code) (937) 255-3636, ext 4703; e-mail: Alan.Johnson@afit.edu